

Universidad Autónoma de Madrid

Escuela Politécnica Superior



TRABAJO DE FIN DE MÁSTER

# Recomendación contextualizada usando una ontología de contexto genérica y de gran escala construida semiautomáticamente a partir de DBpedia

Máster Universitario en Investigación e Innovación en  
Tecnologías de la Información y las Comunicaciones

Por

Nicolás Rodríguez Artigot

bajo la supervisión de

Pedro G. Campos Soto e Iván Cantador Gutiérrez

Madrid, junio de 2017



# Contenidos

<b>Resumen</b>	<b>ix</b>
<b>Abstract</b>	<b>xi</b>
<b>Agradecimientos</b>	<b>xiii</b>
<b>1 Introducción</b>	<b>1</b>
1.1 Motivación .....	1
1.2 Objetivos.....	2
1.3 Contribuciones .....	3
1.4 Estructura del documento.....	3
<b>2 Conceptos preliminares</b>	<b>5</b>
2.1 Sistemas de recomendación .....	5
2.1.1 Sistemas de recomendación basados en contenido.....	7
2.1.2 Sistemas de filtrado colaborativo .....	9
2.1.3 Sistemas de recomendación híbridos .....	12
2.1.4 Sistemas de recomendación contextualizada .....	14
2.1.5 Evaluación de los sistemas de recomendación .....	17
2.2 Web Semántica .....	22
2.2.1 Ontologías y la Web Semántica .....	22
2.2.2 El proyecto Linked Open Data .....	25
2.2.3 DBpedia .....	27
<b>3 Trabajo relacionado</b>	<b>31</b>
3.1 Modelado de contexto .....	31
3.2 Reconocimiento de información contextual en texto .....	36
3.3 Recomendación contextualizada .....	40

<b>4</b>	<b>Solución desarrollada</b>	<b>47</b>
4.1	Taxonomía de contexto a partir de DBpedia.....	47
4.1.1	Categorías principales de la taxonomía.....	47
4.1.2	Construcción automática de la taxonomía .....	49
4.1.3	Edición manual de la taxonomía.....	52
4.1.4	Taxonomía contextual final .....	53
4.2	Anotaciones contextuales en evaluaciones textuales .....	56
4.2.1	Procesamiento del lenguaje natural sobre evaluaciones textuales .....	56
4.2.2	Reconocimiento de categorías e instancias contextuales en evaluaciones textuales .....	57
4.3	Métodos de recomendación contextualizada.....	59
4.3.1	Métodos base de referencia .....	59
4.3.2	Métodos conscientes del contexto.....	61
<b>5</b>	<b>Experimentos</b>	<b>65</b>
5.1	Repositorio de evaluaciones textuales .....	65
5.1.1	Evaluaciones textuales de películas.....	72
5.1.2	Evaluaciones textuales de música .....	76
5.1.3	Evaluaciones textuales de libros.....	79
5.2	Evaluación de las anotaciones de contexto .....	83
5.3	Evaluación de las recomendaciones contextualizadas.....	93
5.3.1	Recomendación sobre películas .....	96
5.3.2	Recomendación sobre música .....	98
5.3.3	Recomendación sobre libros .....	100
<b>6</b>	<b>Conclusiones y trabajo futuro</b>	<b>103</b>
6.1	Conclusiones .....	103
6.2	Trabajo futuro .....	105
	<b>Anexo A Base de datos de la aplicación de evaluación de anotaciones</b>	<b>107</b>
	<b>Referencias</b>	<b>113</b>

# Índice de figuras

Figura 2.1 Clasificación de CARS según Adomavicius et al. (2011) .....	15
Figura 2.2 Estrategias de incorporación y uso de información contextual en CARS según Adomavicius y Tuzhilin (2011) .....	17
Figura 2.3 Web actual contra Web Semántica (Castells, 2003) .....	22
Figura 2.4 Ejemplo de representación RDF para el profesor Iván Cantador .....	23
Figura 2.5 Ejemplo de representación RDFS .....	24
Figura 2.6 Ejemplo de representación OWL .....	25
Figura 2.7 Ejemplo de consulta SPARQL .....	25
Figura 2.8 Linking Open Data cloud diagram 2017-02-20. <a href="http://lod-cloud.net/">http://lod-cloud.net/</a> .....	26
Figura 2.9 Página web de Wikipedia de la Universidad Autónoma de Madrid .....	27
Figura 2.10 Página web en DBpedia para la Universidad Autónoma de Madrid .....	28
Figura 3.1 Representación multidimensional propuesta por Adomavicius et al. (2005) para realizar recomendaciones contextualizadas .....	41
Figura 3.2 Un ítem se separa en dos pseudo-ítems que tienen asociados los distintos valores contextuales y el subconjunto de ratings correspondiente y se integran de nuevo en el sistema (Baltrunas & Ricci, 2014) .....	44
Figura 4.1 Herramienta de construcción y edición de la taxonomía de contexto. ....	52
Figura 4.2 Composición de la información adicional de la taxonomía. ....	54
Figura 4.3 Composición de la taxonomía hasta profundidad cuatro. ....	55
Figura 5.1 Total de reviews con anotaciones y sin anotaciones de contexto. ....	68
Figura 5.2 Total de usuarios con anotaciones y sin anotaciones de contexto. ....	69
Figura 5.3 Total de ítems con anotaciones y sin anotaciones de contexto. ....	70
Figura 5.4 Niveles de matching empleados en la anotación de cada dominio. ....	71
Figura 5.5 Anotaciones distribuidas por dimensión contextual. ....	72
Figura 5.6 Niveles de matching empleados en las reviews de películas. ....	74
Figura 5.7 Anotaciones para las reviews de películas según su dimensión contextual. ....	76
Figura 5.8 Niveles de matching empleados en las reviews de música. ....	77
Figura 5.9 Anotaciones para las reviews de música según su dimensión contextual. ....	79
Figura 5.10 Niveles de matching empleados en las reviews de libros. ....	81
Figura 5.11 Anotaciones para las reviews de libros según su dimensión contextual. ....	83
Figura 5.12 Herramienta de evaluación de anotaciones de contexto. ....	84
Figura 5.13 Anotaciones correctas (Sí/No) por nivel de matching y dominio. ....	87
Figura 5.14 Anotaciones contexto correctas (Sí/No) por nivel de matching y dominio. ....	89
Figura 5.15 Anotaciones correctas (Sí/No) por tipo dimensión contextual y dominio reviews. ....	91

Figura 5.16 Anotaciones contexto correctas (Sí/No) por tipo dimensión contextual y dominio. ....	92
Figura A.1 Diagrama Entidad-Relación de la base de datos para la aplicación de evaluación de anotaciones contextuales.....	107

# Índice de tablas

Tabla 4.1 Categorías de contexto empleadas por algunos modelos ontológicos. ....	48
Tabla 4.2 Categorías raíz de DBpedia elegidas para las dimensiones contextuales.....	50
Tabla 4.3 Estadísticas de la taxonomía genérica de contexto construida. ....	54
Tabla 4.4 Niveles de matching entre palabras de las reviews y las categorías de la taxonomía de contexto. ....	58
Tabla 5.1 Descripción general del dataset de reviews de Amazon.....	66
Tabla 5.2 Números totales de reviews, usuarios e ítems del dataset de Amazon en cada uno de los tres dominios considerados. ....	66
Tabla 5.3 Números totales de reviews, usuarios e ítems empleados en los experimentos en cada uno de los tres dominios considerados.....	67
Tabla 5.4 Total de reviews con anotaciones y sin anotaciones de contexto. ....	67
Tabla 5.5 Total de usuarios con anotaciones y sin anotaciones de contexto.....	68
Tabla 5.6 Total de ítems con anotaciones y sin anotaciones de contexto. ....	69
Tabla 5.7 Números totales de anotaciones en cada dominio y nivel de matching.....	70
Tabla 5.8 Anotaciones según su dimensión contextual. ....	72
Tabla 5.9 Reviews, usuarios e ítems con/sin anotaciones para el dominio de películas.....	73
Tabla 5.10 Niveles de matching empleados en las reviews de películas. ....	73
Tabla 5.11 Categorías contextuales más anotadas en las reviews de películas.....	74
Tabla 5.12 Anotaciones del dominio de películas según su dimensión contextual. ....	75
Tabla 5.13 Reviews, usuarios e ítems para el dominio de música. ....	76
Tabla 5.14 Niveles de matching empleados en las reviews de música.....	77
Tabla 5.15 Categorías contextuales más anotados en las reviews de música. ....	78
Tabla 5.16 Anotaciones del dominio de música según su dimensión contextual.....	79
Tabla 5.17 Reviews, usuarios e ítems para el dominio de libros. ....	80
Tabla 5.18 Niveles de matching empleados en las reviews de libros. ....	80
Tabla 5.19 Categorías contextuales más anotados en las reviews de libros. ....	82
Tabla 5.20 Anotaciones del dominio de libros según su dimensión contextual. ....	82
Tabla 5.21 Tabla resumen tras la evaluación manual de anotaciones. ....	85
Tabla 5.22 Tabla comparativa anotaciones correctas agrupadas según nivel de matching. ....	86
Tabla 5.23 Tabla comparativa anotaciones correctas empleando sólo emparejamientos por coincidencia exacta, instancias de DBpedia y redirecciones a instancias.....	87
Tabla 5.24 Tabla comparativa anotaciones contexto correctas agrupadas según nivel de matching. ....	88

Tabla 5.25 Tabla comparativa anotaciones contextuales correctas empleando sólo emparejamientos por coincidencia exacta, instancias de DBpedia y redirecciones a instancias.....	90
Tabla 5.26 Tabla comparativa anotaciones correctas agrupadas según la dimensión contextual.....	90
Tabla 5.27 Tabla comparativa anotaciones contexto correctas agrupadas según la dimensión contextual.....	92
Tabla 5.28 Características conjuntos de datos para el dominio de películas.....	94
Tabla 5.29 Características conjuntos de datos para el dominio de música. ....	95
Tabla 5.30 Características conjuntos de datos para el dominio de libros. ....	95
Tabla 5.31 Tabla con los métodos de recomendación evaluados.....	96
Tabla 5.32 Resultados en la predicción de rating sobre películas empleando métodos baseline. ....	96
Tabla 5.33 Resultados en la predicción de rating sobre películas empleando métodos conscientes del contexto. ....	97
Tabla 5.34 Resultados en la predicción de rating sobre música empleando métodos baseline.....	98
Tabla 5.35 Resultados en la predicción de rating sobre música empleando métodos conscientes del contexto. ....	99
Tabla 5.36 Resultados en la predicción de rating sobre libros empleando métodos baseline.....	100
Tabla 5.37 Resultados en la predicción de rating sobre libros empleando métodos conscientes del contexto. ....	101
Tabla A.1 Campos de la tabla EXPERTS.....	108
Tabla A.2 Campos de la tabla EXPERTS_ADDITIONAL_ITEM_REVIEWS.....	108
Tabla A.3 Campos de la tabla EXPERTS_ITEM_REVIEWS_CONTEXT .....	109
Tabla A.4 Campos de la tabla EXPERTS_ITEM_REVIEWS_CONTEXT_EVAL_CONTEXT_STATE.....	109
Tabla A.5 Campos de la tabla EXPERTS_ITEM_REVIEWS_CONTEXT_EVAL_STATE.....	110
Tabla A.6 Campos de la tabla ITEM .....	110
Tabla A.7 Campos de la tabla ITEM_REVIEWS .....	110
Tabla A.8 Campos de la tabla ITEM_REVIEWS_CONTEXT .....	111
Tabla A.9 Campos de la tabla REVIEWER.....	111
Tabla A.10 Campos de la tabla TAXONOMY .....	112
Tabla A.11 Campos de la tabla TAXONOMY_MATCHING_LEVEL .....	112



# Glosario

**CARS** *Context-Aware Recommender Systems*

Sistemas de recomendación contextualizados, que además de preferencias de usuario explotan información contextual (e.g., tiempo, lugar, tiempo meteorológico, compañía social, estado de ánimo) para generar recomendaciones personalizadas.

**OWL** *Web Ontology Language*

Lenguaje de marcado para publicar y compartir datos usando ontologías en la WWW.

**RDF** *Resource Description Framework*

Infraestructura XML que permite la codificación, intercambio y reutilización de meta-datos estructurados.

**RDFS** *RDF Schema*

Conjunto de clases con ciertas propiedades que usan el modelo de representación de datos RDF para proporcionar elementos básicos de descripción de ontologías.

**SPARQL** *SPARQL Protocol and RDF Query Language.*

Lenguaje de consulta de base de datos semántico capaz de recuperar y manipular datos almacenados en formato RDF.

**SQL** *Structured Query Language*

Lenguaje de propósito especial y dominio específico usado para manejar una base de datos relacional.

**URI** *Uniform Resource Identifier*

Cadena de caracteres usada para identificar unívocamente un recurso, permitiendo la interacción con representaciones del recurso en una red, típicamente la WWW, usando protocolos específicos.

**XML** *Extensible Markup Language*

Lenguaje de marcado con un conjunto de reglas para codificar documentos en un formato legible tanto por humanos como programas informáticos.

**W3C** *World Wide Web Consortium*

Principal organización internacional para la estandarización de la WWW.

**WWW** *World Wide Web (también conocida como la Web)*

Espacio de información donde documentos y otros recursos web son identificados mediante Localizadores de Recursos Uniformes (*Uniform Resource Locators, URLs*), inter-conectados mediante enlaces de hipertexto, accesibles vía internet.



# Resumen

Los sistemas de recomendación son un tipo específico de filtrado de información que presenta a un usuario recursos (películas, música, libros, noticias, páginas web, etc.) que le pueden ser relevantes, atendiendo a su perfil de preferencias (gustos, intereses, necesidades), sin que tenga que realizar búsquedas explícitas de dichos recursos.

Junto con sus preferencias, también se puede utilizar información sobre el contexto (e.g., tiempo, localización, compañía) del usuario para obtener mejores predicciones de relevancia. Los sistemas que la explotan se han denominado sistemas de recomendación conscientes del contexto (del inglés *context-aware recommender systems*).

Habitualmente en la generación de recomendaciones contextualizadas se parte de un conjunto limitado de atributos y valores contextuales, que además suelen ser indicados de manera explícita por el usuario. Abordando esta situación, en este trabajo se propone la extracción de información contextual existente en valoraciones textuales (*reviews*) escritas por los usuarios sobre diversos recursos.

Para modelar el contexto se ha construido semi-automáticamente una taxonomía de contexto genérica construida a partir de entidades extraídas de DBpedia –la base de conocimiento estructurada de la enciclopedia online Wikipedia–, que contempla cuatro dimensiones contextuales principales: el tiempo, la localización, el entorno y la compañía del usuario. En total, la taxonomía cuenta con 554 categorías contextuales, las cuales se han enriquecido lexicalmente con instancias de DBpedia, así como con sinónimos extraídos del diccionario WordNet para las categorías e instancias.

A partir de esta taxonomía enriquecida se ha llevado a cabo un proceso automático de anotación contextual mapeando las categorías contextuales con palabras existentes en las reviews, utilizando para ello la información adicional de la taxonomía (i.e., instancias y sinónimos). Para validar estas anotaciones, se ha desarrollado una aplicación software de evaluación que asiste la revisión manual de las mismas. Además, se ha realizado un análisis detallado de estas anotaciones sobre tres dominios de datos, a saber, películas, música y libros.

Por último, estas anotaciones han sido evaluadas adicionalmente mediante distintos métodos de recomendación contextualizada, comparando sus resultados con los de otros métodos de referencia que no utilizan información contextual. Los resultados obtenidos son ciertamente prometedores, ya que el uso de la información contextual anotada ha permitido en los experimentos realizados obtener mejores recomendaciones frente a los métodos de referencia.

**Palabras clave:** taxonomía de contexto, extracción de contexto, sistemas de recomendación conscientes del contexto, reviews de usuario, DBpedia.



# Abstract

Recommender systems are a particular type of information filtering that presents resources (movies, music, books, news, web pages, etc.) of potential relevance for a user according to her preferences (tastes, interests, needs), without the need of explicit searches of such resources.

Along with her preferences, the user's context (e.g., time, location, social companion) can also be used to obtain better relevance predictions. The systems that exploit such information are called context-aware recommender systems.

In general, the generation of contextualized recommendations is conducted in terms of a limited set of contextual attributes and values, which usually are explicitly provided by the users, a fact that limits the applicability of this type of recommendations. Addressing this situation, in this work we propose to extract contextual information from textual reviews written by users, expressing their opinions and experiences with the resources.

To model context we semi-automatically built a generic context taxonomy with semantic entities extracted from DBpedia –the structured knowledge base of the Wikipedia online encyclopedia–, covering four context dimensions: time, location, environment and social companion. In total, the taxonomy has 554 context categories, which were lexically enriched with DBpedia instances, and synonyms of the categories and instances obtained from the WordNet lexical database.

Exploiting the enriched taxonomy, we performed an automatic context annotation process on the user reviews, mapping contextual categories with words in the reviews, with the help of the additional information in the taxonomy (instances and synonyms). To validate the annotations, we developed a software application that assisted their manual revision, and conducted a detailed analysis of the annotations in three domains, namely movies, music and books.

Finally, we evaluated the context annotations empirically comparing several context-aware recommender systems with baseline recommendation approaches that do not use contextual information. The achieved results are promising, since the exploitation of the automatically-generated contextual information has led to better recommendations than the generated by the baselines.

**Keywords:** context taxonomy, context extraction, context-aware recommender systems, user reviews, DBpedia.



## Agradecimientos

En primer lugar, me gustaría agradecer el trabajo llevado a cabo por mis dos tutores, Iván Cantador Gutiérrez y Pedro G. Campos Soto, por darme la oportunidad de formar parte de este proyecto, por haberme acompañado durante todo este periodo y por vuestra labor desempeñada semana tras semana para que esto saliese adelante. Muchas gracias por todo.

También quería dar las gracias a mi familia y en particular a mis padres por todo el apoyo recibido durante este tiempo, día tras día, de manera constante, para poder completar este trabajo. Muchas gracias de todo corazón.

Por último, agradecer a todo aquel que con un café/té de por medio o incluso sin ser consciente, con un simple '*¿Qué tal llevas el TFM?*' has hecho que este trabajo sea más ligero. Una parte de este trabajo es gracias a ti.

Nico





## Capítulo 1

# Introducción

### 1.1 Motivación

Los sistemas de recomendación representan un tipo específico de filtrado de información que presenta al usuario recursos (películas, música, libros, noticias, páginas web, etc.) que le pueden ser relevantes, atendiendo a su perfil de preferencias (gustos, intereses, necesidades), sin necesidad de que realice búsquedas explícitas de dichos recursos. Desde mediados de los 90, son objeto de investigación activa y se han integrado con éxito en numerosos sitios de comercio electrónico y ocio en la Web tales como Amazon<sup>1</sup>, Netflix<sup>2</sup>, YouTube<sup>3</sup>, iTunes<sup>4</sup> y Last.fm<sup>5</sup>.

Además de las preferencias de usuario, los sistemas de recomendación pueden hacer uso de información adicional para mejorar sus predicciones de relevancia. En concreto, de especial interés son aquellas aproximaciones enfocadas a explotar información sobre el contexto actual del usuario: el tiempo (e.g., día de la semana, franja horaria del día, estación del año), la localización (e.g., en casa, en el trabajo, en el coche), el estado del ánimo, la compañía del usuario (e.g., solo, en pareja, en familia, con amigos), entre otros.

En la literatura se puede encontrar un gran número de trabajos sobre modelos de recomendación conscientes del contexto (del inglés *context-aware recommender*

---

<sup>1</sup> Amazon, plataforma de comercio electrónico, <http://www.amazon.com>

<sup>2</sup> Netflix, proveedor de *streaming* multimedia en internet, <http://www.netflix.com>

<sup>3</sup> YouTube, sitio web dedicado a compartir videos, <https://www.youtube.com>

<sup>4</sup> iTunes, tienda de contenido multimedia de Apple, <http://www.apple.com/itunes>

<sup>5</sup> Last.fm, radio on-line, <https://www.last.fm>

*systems*). Sin embargo, en la gran mayoría de ellos se parte de un conjunto limitado y ad-hoc de atributos y valores contextuales, que en general han de ser descritos de manera explícita por el usuario al usar el sistema.

El objetivo general de este trabajo es obtener de manera automática a partir de fuentes textuales (reviews) información útil para realizar recomendaciones contextualizadas, empleando una ontología de contexto genérica y de gran escala para la anotación automática de las reviews.

Para ello, se plantea necesario el de definir y construir una ontología y base de conocimiento estructurada sobre contexto, que aborde diversos dominios. En particular, se propone extraer dicha ontología a partir de DBpedia, la cual es un repositorio web, abierto y gratuito, que contiene información estructurada proveniente de la Wikipedia. Como primera aproximación, se pretende que la construcción de la ontología siga un proceso semi-automático, a través de una herramienta ad hoc que asista la obtención automática de datos de DBpedia y su posterior filtrado manual.

Una vez construida la ontología, se realizará un procesamiento de un conjunto de datos con evaluaciones textuales por parte de usuarios acerca de distintos productos como películas, música o libros de la Web de Amazon, y se llevará a cabo su anotado automático con dimensiones y valores contextuales, de modo que se generará un conjunto de datos de gran escala con evaluaciones de usuario contextualizadas. Este proceso será cíclico, siendo necesario remodelar la ontología y el anotado de las evaluaciones por parte de los usuarios hasta generar un conjunto de datos válido.

Una vez preparados los datos, se llevará a cabo una evaluación de distintos algoritmos de recomendación conscientes del contexto para comparar su rendimiento sobre el conjunto de datos anotado previamente.

## 1.2 Objetivos

Los objetivos concretos que persigue el TFM son los siguientes:

- Modelar y construir una ontología de contexto adecuada para almacenar información contextual asociada a un conjunto de datos dado a partir de DBpedia, la base de conocimiento estructurada asociada a Wikipedia. Esta estructura debe ser suficientemente grande y genérica para poder trabajar con diversos dominios y aplicaciones conscientes del contexto.
- Anotar un conjunto de evaluaciones textuales de usuarios con conceptos e instancias existentes en la ontología de contexto construida utilizando la librería CoreNLP de la Universidad de Stanford para llevar a cabo el procesamiento de lenguaje natural de los textos y posterior anotación.

- Evaluar varios algoritmos de recomendación contextualizada para comparar su rendimiento utilizando los conjuntos de datos anotados con información de contexto.

## 1.3 Contribuciones

En este trabajo se encuentran las siguientes contribuciones:

- Una ontología de contexto genérica y de gran tamaño extraída a partir de DBpedia, que puede ser usada para distintos dominios y aplicaciones conscientes del contexto.
- Un repositorio de anotaciones contextuales sobre un conjunto de reviews del estado del arte, generadas utilizando la taxonomía de contexto genérica construida.
- Una herramienta que permite llevar a cabo la evaluación y modificación manuales de anotaciones de contexto existentes en reviews.
- La evaluación de distintos algoritmos de recomendación contextualizada frente a algoritmos de recomendación tradicionales utilizando las correspondientes anotaciones contextuales.

Parte de este trabajo se presenta en el siguiente trabajo de investigación:

- Campos, P. G., Rodríguez-Artigot, N., Cantador, I. 2017. **Extracting context data from user reviews for recommendation: A Linked Data approach**. Submitted to the *1<sup>st</sup> Workshop on Recommendation in Complex Scenarios (ComplexRec 2017)*, at the 11<sup>th</sup> ACM Conference on Recommender Systems (RecSys 2017). 31<sup>st</sup> August, 2017. Como, Italy.

## 1.4 Estructura del documento

Este trabajo está estructurado de la siguiente manera:

- **Capítulo 2. Conceptos preliminares.** En este capítulo se presentan conceptos fundamentales que permiten entender mejor este trabajo, como es una visión general de los sistemas de recomendación (e.g., filtrado basado en contenido y filtrado colaborativo, recomendación contextualizada y evaluación de recomendaciones), y de la Web Semántica (e.g., ontologías, Linked Open Data y DBpedia)
- **Capítulo 3. Trabajo relacionado.** En este capítulo se presenta una revisión del Estado del Arte en los tres pilares en los que se encuadra este trabajo: el modelado de contexto, la extracción de información contextual a partir de un texto, y los sistemas de recomendación contextualizada.

- **Capítulo 4. Solución desarrollada.** En este capítulo se presenta la solución desarrollada para la construcción semi-automática de la taxonomía genérica de contexto, la extracción de información contextual de las anotaciones textuales (reviews) escritas por usuarios, y la posterior explotación de las anotaciones contextuales por distintos métodos de recomendación contextualizada.
- **Capítulo 5. Experimentos.** En este capítulo se detallan los experimentos llevados a cabo, como son el análisis y la evaluación de las anotaciones contextuales obtenidas automáticamente a partir de las reviews de usuario, y la evaluación de métodos de recomendación básicos y contextualizados usando las anotaciones contextuales obtenidas.
- **Capítulo 6. Conclusiones y trabajo futuro.** En este capítulo se exponen las conclusiones obtenidas tras la realización de este trabajo, así como distintas líneas de trabajo futuro.

## Capítulo 2

# Conceptos preliminares

En este capítulo se presenta una serie de conceptos preliminares, necesarios para establecer el marco de referencia en el que se encuadra esta investigación. Estos conceptos en primer lugar tratan los sistemas de recomendación, ya que representan el área de investigación que engloba el tema fundamental del trabajo: la recomendación contextualizada. Posteriormente, versan sobre la Web Semántica y algunas de sus tecnologías principales, necesarias para entender qué son las ontologías y DBpedia, representación y base del conocimiento sobre los que se ha desarrollado la aproximación de recomendación contextualizada propuesta.

### 2.1 Sistemas de recomendación

Un **sistema de recomendación** se puede definir como una herramienta software que ayuda a un usuario a encontrar en una colección de ítems (películas, música, libros, o productos en general), aquellos que le son relevantes atendiendo a sus preferencias, i.e., gustos, intereses o necesidades. La colección de ítems suele ser lo demasiado extenso como para que el usuario lo revise de forma manual, e.g., el catálogo de productos de Amazon, o la base de datos de películas y series de televisión de Netflix.

Los sistemas de recomendación se han consolidado como un área de investigación muy relevante, que han evolucionado desde su aparición en primeros trabajos científicos a mediados de los años 90, cuando los investigadores pusieron el foco en problemas de filtrado colaborativo y explotación de *ratings* (i.e., valoraciones numéricas) explícitos de ítems dados por usuarios.

Así, en uno de las primeras revisiones del estado del arte en sistemas de recomendación, Adomavicius y Tuzhilin (2005) definen el problema de recomendación como la

estimación de ratings para aquellos ítems que un usuario objetivo todavía no ha explorado o utilizado. Para estimar los ratings de esos ítems, se emplea información conocida por el sistema, siendo la más común la de ratings previos del usuario sobre otros ítems. Posteriormente, una vez estimados los ratings para los ítems desconocidos, el sistema sugeriría al usuario aquellos con los ratings estimados más altos.

Lo anterior se puede formalizar como sigue. Sea  $U$  el conjunto de usuarios del sistema,  $I$  el conjunto de ítems del catálogo del sistema, y  $f$  una determinada función de utilidad  $f(u, i)$  que determina cómo de útil (relevante) es el ítem  $i$  para el usuario  $u$ :

$$f: U \times I \rightarrow R$$

donde  $R$  representa un conjunto ordenado de valores, e.g., una serie de números enteros positivos, o un determinado rango de números reales. El objetivo de la recomendación es que para cada usuario  $u \in U$ , se encuentre un ítem  $i^* \in I$  desconocido por el usuario, tal que maximice esa función de utilidad:

$$i_u^* \in I = \arg \max_{i \in I} f(u, i)$$

De forma habitual, la utilidad de un determinado ítem se suele representar a modo de rating, e.g., valores del 1 al 5, indicando así el grado con el que al usuario valora el ítem.

El problema central de los sistemas de recomendación es que la función de utilidad  $f$  no está definida para todo el conjunto  $U \times I$ , sino sólo para una pequeña parte del mismo, obtenida a partir de interacciones previas entre pares usuario-ítem en el sistema. Por tanto, la labor del sistema consiste en extrapolar los valores de esa función de utilidad para todo el conjunto  $U \times I$ .

Los ratings estimados para los pares usuario-ítem desconocidos pueden obtenerse de diversas maneras. En particular, se puede establecer una **categorización de los sistemas de recomendación** en función de los métodos empleados para obtener esos ratings (Adomavicius y Tuzhilin, 2005):

- **Sistemas de recomendación basados en contenido (CB)**, que consideran rasgos y características de los ítems (palabras clave, categorías, etc.), de forma que el usuario obtenga ítems recomendados con características similares a los ítems que prefirió en el pasado.
- **Sistemas de filtrado colaborativo (CF)**, que tratan los ítems de manera opaca (sin considerar características propias), y emplean las preferencias de otros usuarios con rasgos similares para recomendar ítems al usuario actual. En este contexto, la similitud entre usuarios se suele establecer a partir de correlaciones y patrones de ratings.

- **Sistemas de recomendación híbridos (HYB)**, que siguen alguna estrategia para combinar métodos de recomendación basados en contenido y de filtrado colaborativo.

En las siguientes subsecciones se detallan algunos de los métodos más populares para cada uno de los tipos de sistemas de recomendación anteriores.

### 2.1.1 Sistemas de recomendación basados en contenido

Los **sistemas de recomendación basados en contenido** (CB) buscan encontrar ítems similares a aquellos preferidos (e.g., mejor valorados/puntuados) por el usuario objetivo en el pasado. De esta forma, la función de utilidad  $f(u, i)$ , que determina como de relevante es el ítem  $i$  para el usuario  $u$ , se puede calcular empleando la utilidad  $f(u, i')$  de aquellos ítems  $i'$  que son “similares” al ítem  $i$ .

De este modo, la dificultad del problema radica en definir una representación de los ítems, así como una medida de similitud entre ítems, que sean efectivas. Para ello, se emplean rasgos o características de los propios ítems; por ejemplo, si los ítems son películas, se puede emplear información como el director, los actores, y los géneros de las películas para conformar el contenido o **perfil de ítem**  $Contenido(i)$ .

Para la construcción del perfil habitualmente se emplean técnicas del área de Recuperación de Información, ya que se puede entender que los ítems están representados como documentos de texto, donde sus características o rasgos vienen dados mediante palabra claves o *keywords*. Con todo eso, el perfil de ítem,  $Contenido(i)$ , no sería otra cosa sino un vector de pesos  $w_k$  asociados a cada una de las palabras clave dentro del ítem:

$$Contenido(i) = (w_{1i}, w_{2i}, \dots, w_{ki})$$

De entre todas las alternativas propuestas para calcular los pesos  $w_k$ , una de las más empleadas es la conocida como TF-IDF (Salton, 1989), que emplea la combinación de los factores de Frecuencia de Términos o *term frequency* (TF), y la Frecuencia Inversa de Documento o *Inverse Document Frequency* (IDF).

El factor TF se define como la frecuencia normalizada de la palabra  $k_i$  dentro del documento  $d_j$ :

$$TF_{ij} = \frac{\text{Numero apariciones de } k_i \text{ en } d_j}{\max_z \text{ Numero apariciones de } k_z \text{ en } d_j}$$

En el caso de ítems, las apariciones de una característica dentro de un ítem pueden ser binarias (e.g., si una película pertenece o no a un determinado género), o numéricas, como en el caso de los textos, el número de veces que aparece una palabra dentro del documento. En este último caso, el número de apariciones de una característica se normaliza por el valor máximo de apariciones obtenido para cualquiera de las

características dentro del ítem. De esta forma, las características que más aparezcan en el ítem, se entienden como las más representativas del mismo.

El factor IDF representa en cierto modo el extremo opuesto a TF: pretende capturar aquellas características que son más discriminativas, i.e., menos frecuentes, dentro del conjunto de ítems. Por eso se define como:

$$IDF_i = \log \frac{\text{Número total de ítems}}{\text{Ítems donde esta presente la característica } k_i}$$

De este modo, si una característica está presente en un gran número de ítems, su valor IDF será pequeño, entendiendo que esa característica no es muy idónea como rasgo identificativo del ítem.

Finalmente, el peso específico de cada una de las características del ítem se calcula empleando una combinación de esos dos factores:

$$w_{ij} = TF_{ij} \times IDF_i$$

De forma similar al perfil del ítem, los sistemas de recomendación basada en contenido emplean una representación del usuario, denominada **perfil de usuario**, entendida como un conjunto de rasgos o características que describen las preferencias del usuario. Este perfil de usuario también puede representarse como un vector de pesos  $w_c$  que representen la importancia de cada una de las características modeladas en *ContenidoPerfil(u)*:

$$\text{ContenidoPerfil}(u) = (w_{1u}, w_{2u}, \dots, w_{cu})$$

Teniendo lo anterior en cuenta, la función de utilidad  $f(u, i)$  del ítem  $i$  para el usuario  $u$ , se calcula como un valor numérico *score* teniendo en cuenta tanto el perfil del ítem, como el perfil del usuario:

$$f(u, i) = \text{score}(\text{ContenidoPerfil}(u), \text{Contenido}(i))$$

Para calcular este *score* pueden emplearse técnicas heurísticas como por ejemplo la métrica de similitud del coseno (Baeza-Yates & Ribeiro-Neto, 1999), o modelos de aprendizaje automático, como por ejemplo los clasificadores Bayesianos, las técnicas de *clustering*, los árboles de decisión y las redes neuronales (Pazzani & Billsus, 1997).

En cualquier caso, como apuntan Adomavicius y Tuzhilin (2005), los sistemas de recomendación basados en contenido tienen una serie de **limitaciones**:

- **Análisis limitado del contenido.** Los sistemas CB están limitados por las características asociadas a los ítems, ya que estas se definen de forma explícita. Esto implica que si se quiere emplear esto de forma óptima, esas características deban ser extraídas de forma automática, lo cual es sencillo para algunos tipos de ítem (e.g., texto), pero complejo o no factible en otros (e.g., música). Además, otro punto derivado de esta limitación, es que si dos ítems están



formados por el mismo conjunto de características, desde el punto de vista del sistema, esos ítems serían idénticos e indistinguibles.

- ***Sobre-especialización de las recomendaciones.*** Al vincular el *score* de utilidad al perfil del usuario, i.e., a los ítems preferidos por el usuario, el sistema sólo recomendará ítems con unas características muy similares a los que el usuario ya conoce, de forma que se tenga una diversidad baja de entre los ítems recomendados, cuando lo ideal podría ser tener un rango de recomendaciones bastante diferentes, a la par que relevantes para el usuario.
- ***Problema de arranque en frío (cold start) para un usuario nuevo.*** Cuando un usuario es nuevo en el sistema y ha valorado ningún o un número reducido de ítems, el sistema es incapaz de ofrecer recomendaciones precisas. Por tanto, para que los sistemas CB funcionen de manera efectiva, se requiere disponer de información suficiente del usuario.

### 2.1.2 Sistemas de filtrado colaborativo

La principal diferencia de los **sistemas de filtrado colaborativo (CF)** respecto a los sistemas de recomendación basados en contenido, es que para los primeros el cálculo de la relevancia que un ítem puede tener para el usuario se emplean valoraciones de otros ítems dadas por usuarios afines, mientras que en los segundos sólo se emplean las valoraciones de ítem del propio usuario. De esta forma, se acaba con la sobre-especialización que poseen las recomendaciones basadas en contenido, y se proporciona una mayor diversidad de ítems novedosos.

En los algoritmos de filtrado colaborativo, la relevancia de un ítem se entiende como el *rating* que tiene ese ítem dado por el usuario. Así, la finalidad de estos algoritmos es la de estimar el rating que un usuario  $u$  daría a ítems  $i$ , para después poder recomendar aquellos ítems con mayor rating estimado.

Los sistemas de filtrado colaborativo son bastante populares y son utilizados habitualmente, e.g., en plataformas de comercio electrónico como Amazon (Linden et al., 2003), empleando en muchos casos aproximaciones del famoso **algoritmo de vecinos próximos**, o *k-Nearest Neighbours*, kNN, basado en heurísticas para determinar los  $k$  usuarios más afines al usuario objetivo (kNN basado en usuario), o los  $k$  ítems con patrones de rating más similares al ítem cuyo rating se está estimando (kNN basado en ítem).

Más específicamente, los algoritmos kNN basados en usuario (Resnick et al., 1994), estiman el rating  $\hat{r}_{u,i}$  de un ítem  $i$  empleando los ratings de aquellos  $k$  usuarios (vecinos) más similares al usuario objetivo  $u$ , entendiendo por similares aquellos usuarios que hayan valorado otros ítems de forma similar a cómo los valoró  $u$ . De

forma genérica, el rating estimado para el ítem  $i$  por el usuario  $u$  se calcula como una función de agregación empleando los ratings  $r_{u',i}$  de esos  $k$  vecinos  $u'$ :

$$\hat{r}_{u,i} = \text{aggr}_{u' \in k \text{ vecinos } u} r_{u',i}$$

Para determinar los  $k$  vecinos se pueden emplear diferentes métodos, siendo los más utilizados los métodos basados en correlación y los métodos basados en la similitud por coseno. Antes de definirlos de manera formal, es necesario considerar que  $I_u = \{i \in I, r_{ui} \neq 0\}$  es el conjunto de los ítems valorados por el usuario  $u$ ;  $I_v = \{i \in I, r_{vi} \neq 0\}$  es el conjunto de los ítems valorados por el usuario  $v$ ; e  $I_{uv} = \{i \in I, r_{ui} \neq 0, r_{vi} \neq 0\}$  es el conjunto de los ítems valorados por ambos usuarios  $u$  y  $v$ .

Teniendo en cuenta estas consideraciones, para los métodos basados en correlación se emplea el coeficiente de correlación de Pearson (PC) (Resnick et al., 1994), definido como:

$$\text{sim}_{PC}(u, v) = \frac{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)^2 \sum_{i \in I_{uv}} (r_{vi} - \bar{r}_v)^2}}$$

donde  $\bar{r}_u$  la media de los ratings dados por  $u$ .

En cuanto a los métodos basados en similitud por coseno (COS), dos usuarios  $u$  y  $v$  son representados como dos vectores en los que se tienen en cuenta los ítems valorados por ambos usuarios. De esta forma, se tiene un espacio de dimensión  $m$ , donde  $m$  se define formalmente como  $m = |I_{uv}|$ , y la similitud entre ambos vectores se calcula como el coseno del ángulo formado por ellos:

$$\text{sim}_{COS}(u, v) = \cos(r(u), r(v)) = \frac{\sum_{i \in I_{uv}} r_{ui} r_{vi}}{\sqrt{\sum_{i \in I_u} r_{ui}^2 \sum_{i \in I_v} r_{vi}^2}}$$

Una vez que se tienen los usuarios más similares al usuario actual, se emplea una función de agregación para calcular el valor del rating  $\hat{r}_{u,i}$  estimado para el ítem  $i$  por parte del usuario  $u$  con los distintos ratings proporcionados por esos usuarios, siendo la suma ponderada la aproximación más utilizada (Adomavicius & Tuzhilin, 2005):

$$\hat{r}_{u,i} = c \sum_{u' \in k \text{ vecinos } u} \text{sim}(u, u') \times r_{u',i}$$

donde  $c$  es un factor de normalización, que habitualmente se calcula como  $c = 1 / \sum_{u' \in k \text{ vecinos } u} |\text{sim}(u, u')|$ .

Por otro lado, los algoritmos kNN basados en ítem (Sarwar et al., 2001), en vez de recomendar ítems a partir de usuarios similares al usuario objetivo  $u$ , recomiendan ítems similares a los ítems que  $u$  prefirió (asignándoles mayores ratings) en el pasado. Para ello, como  $k$  vecinos del ítem  $i$  se seleccionan aquellos que fueron valorados de forma más similar a  $i$ . Posteriormente, mediante una función de agregación se estima el

rating para  $i$ , empleando los ratings que el usuario  $u$  asignó a los ítems más similares a  $i$ , y el grado de similitud entre ambos.

Para determinar el grado de similitud de dos ítems se parte de la premisa de que dos ítems son similares si de forma habitual son valorados de la misma manera por los distintos usuarios. Así, el primer paso para calcular la similitud entre dos ítems es aislar aquellos usuarios que han valorado ambos ítems, y después emplear alguna medida de similitud (parecidas a las empleadas para calcular la similitud entre usuarios) para determinar el grado de similitud entre los ítems. Antes de definir estas funciones de similitud de manera formal, se considera que  $U_i = \{u \in U, r_{ui} \neq 0\}$  es el conjunto de los usuarios que han valorado el ítem  $i$ ;  $U_j = \{u \in U, r_{uj} \neq 0\}$  es el conjunto de los usuarios que han valorado el ítem  $j$ ; y  $U_{ij} = \{u \in U, r_{ui} \neq 0, r_{uj} \neq 0\}$  es el conjunto de los usuarios que han valorado ambos ítems  $i$  y  $j$ .

En cuanto a las maneras de calcular la similitud entre ítems, Sarwar et al. (2001) presentan tres métodos diferentes: un primer método basado en la similitud por coseno, un segundo método basado en el coeficiente de correlación de Pearson, y un tercer método denominado por coseno ajustado.

La fórmula para determinar la similitud entre dos ítems empleando el método de similitud por coseno (COS) trata los ítems como si fueran vectores de  $m$  dimensiones, donde  $m = |U_{ij}|$ :

$$\text{sim}_{\text{COS}}(i, j) = \cos(r(i), r(j)) = \frac{\sum_{i \in U_{ij}} r_{ui} r_{uj}}{\sqrt{\sum_{u \in U_{ij}} r_{ui}^2 \sum_{u \in U_{ij}} r_{uj}^2}}$$

De forma análoga, para calcular la similitud entre ítems empleando el coeficiente de correlación de Pearson (PC), es necesario emplear exclusivamente aquellos usuarios que valoraron ambos ítems, quedando el cálculo de la siguiente manera:

$$\text{sim}_{\text{PC}}(i, j) = \frac{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_i)(r_{uj} - \bar{r}_j)}{\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_i)^2 \sum_{u \in U_{ij}} (r_{uj} - \bar{r}_j)^2}}$$

Calcular la similitud entre ítems empleando el método de similitud por coseno tiene una desventaja, y es que no se está teniendo en cuenta la diferencia que puede existir en el rango de ratings que emplean los usuarios para valorar los ítems. El método del coseno ajustado (AC) resta el rating promedio del usuario en lugar de restar el rating promedio del ítem:

$$\text{sim}_{\text{AC}}(i, j) = \cos(r(i), r(j)) = \frac{\sum_{i \in U_{ij}} (r_{ui} - \bar{r}_u)(r_{uj} - \bar{r}_u)}{\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_u)^2 \sum_{u \in U_{ij}} (r_{uj} - \bar{r}_u)^2}}$$

En (Sarwar et al., 2001) se discute que las técnicas basadas en ítem obtienen mejores resultados que las basadas en usuario cuando el número de usuarios es mucho mayor

que el número de ítems empleados, i.e.,  $|U| \gg |I|$ . Éste el caso de por ejemplo Amazon.com, que según describe (Linden et al., 2003) usa kNN basado en ítem con un número mucho mayor de usuarios que ítems.

Las técnicas basadas en ítem tienen a su vez las siguientes **ventajas**:

- Matriz de similitudes más pequeña, para así poder mantenerla mejor en memoria RAM.
- Mayor tasa de solapamiento entre ítems, por tanto, similitudes más significativas.
- Menor necesidad de actualización, ya que los ítems suelen ser más estables que los usuarios.

Sin embargo, tal y como apuntan Adomavicius y Tuzhilin (2005), los sistemas de filtrado colaborativo también tienen una serie de **desventajas**:

- **Problema de arranque en frío para un usuario nuevo.** Como los sistemas de filtrado colaborativo se basan en calcular similitudes entre usuarios o ítems, para que estos tengan un funcionamiento aceptable es necesario que se dispongan de ratings suficientes acerca de las preferencias de un usuario. De este modo, un usuario recién ingresado en el sistema es susceptible de no obtener recomendaciones relevantes. Este problema también se denomina en la literatura como *cold start* o arranque en frío.
- **Problema de arranque en frío para un ítem nuevo.** Para que un ítem sea recomendado por un sistema de filtrado colaborativo, es necesario que cuente con un número aceptable de ratings, por tanto, si un ítem no ha sido valorado por suficientes usuarios, dicho ítem es susceptible de que sea recomendado por el sistema en comparación con otros más populares.
- **Escasez y dispersión de ratings de usuario.** En filtrado colaborativo el número de ratings disponibles es muy pequeño en comparación con el número de ratings que se pretenden estimar, lo que dificulta la labor de obtener estimaciones precisas.

### 2.1.3 Sistemas de recomendación híbridos

Para paliar desventajas particulares de los métodos de recomendación basados en contenido y de los métodos de filtrado colaborativo, e.g., la sobre-especialización y el problema del nuevo ítem, es común desarrollar **sistemas de recomendación híbridos** que combinan de alguna manera ambas aproximaciones. Burke (2002) clasifica las estrategias de recomendación híbridas en las siguientes categorías:

- **Weighted hybridization.** Este tipo de recomendaciones consiste en combinar los resultados de varias técnicas de recomendación con el fin de producir una

única recomendación, por ejemplo, empleando una combinación lineal de los distintos scores.

- ***Switching hybridization.*** Bajo este tipo de recomendación, el sistema analiza la situación actual y elige entre las distintas técnicas de recomendación la más apropiada, y hace uso de ella. Es necesario determinar ese criterio para elegir la técnica que debe aplicarse bajo cada situación, por lo que eso añade un nivel más de parametrización.
- ***Mixed hybridization.*** Este tipo de estrategia de hibridación presenta al mismo tiempo los resultados de los distintos sistemas de recomendación, por lo que se ejecutan todas las técnicas de recomendación disponibles y se presentan todos los resultados.
- ***Cascade hybridization.*** Este tipo de hibridación emplea en primer lugar algún método de recomendación para producir un determinado ranking de candidatos que sirve como entrada para un segundo método de recomendación que redefine ese ranking, y genera así un nuevo ranking. No obstante, aquellos ítems que se encuentren bien diferenciados o que tengan un valor de rating muy bajo tras la salida del primer método, no es necesario que se empleen en el segundo, por lo que se aumenta la eficiencia del sistema al no tener que procesar de nuevo todos los ítems.
- ***Feature combination-based hybridization.*** Este tipo de sistemas consiste en emplear las características de una técnica de recomendación y emplearlas como entrada para otra técnica. Por ejemplo, emplear la información colaborativa como si fuese características asociadas a cada dato, y emplear un sistema de recomendación basado en contenido sobre esas ‘nuevas’ características.
- ***Feature augmentation-based hybridization.*** Este tipo de sistemas emplea la salida de una técnica de recomendación como características para la entrada de la siguiente técnica de recomendación. Por ejemplo aplicando métodos de filtrado colaborativo para descubrir los ítems relacionados entre sí, y emplear esa información como características propias de los ítems dentro de un método basado en contenido.
- ***Meta level hybridization.*** Esta forma de recomendación consiste en emplear el modelo generado por un sistema de recomendación como la entrada de otro. De esta forma si por ejemplo se combinan técnicas basadas en contenido y colaborativas, se obtendrá un sistema donde el modelo aprendido tenga en cuenta tanto los intereses del usuario, como el mecanismo colaborativo correspondiente.

### 2.1.4 Sistemas de recomendación contextualizada

La mayoría de los sistemas de recomendación tradicionales se basan en recomendar los ítems más relevantes a los usuarios a partir de las preferencias de los usuarios, pero sin tener en cuenta información contextual que les puede influenciar como puede ser el tiempo, el lugar o la compañía del usuario, a la hora de elegir un restaurante al que ir a cenar o de una película que ver. En otras palabras, sólo explotan información extraída a partir de las interacciones usuario-ítem.

Sin embargo, en muchas aplicaciones y dominios, como por ejemplo al recomendar paquetes vacacionales, productos en una tienda online o películas, considerar las preferencias de usuario de manera aislada puede que sea insuficiente, por lo que es importante incorporar información contextual a la toma de decisiones (Adomavicius et al., 2005). Por ejemplo, entre semana un usuario puede que prefiera por la mañana leer noticias actuales y por la tarde leer informes bursátiles, mientras que los fines de semana es posible que prefiera leer artículos sobre cultura y ocio. Análogamente, a la hora de ver una película, un usuario puede que prefiera diferentes géneros y tipos de películas si está solo, con su pareja o acompañado por amigos.

Para comprender los conceptos que tratan sobre el modelado de contexto es necesario primero definir **contexto**. En literatura actual, una de las definiciones de contexto más empleadas en Computación Ubicua es la siguiente (Dey 2001):

*Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.*

Como se puede observar, esta definición es genérica, pero es sencilla e indica que puede entenderse como *contexto* a toda aquella información que sirva para identificar el entorno de los elementos relevantes en una aplicación (informática), lo cual es válido para sistemas de recomendación.

En la misma línea, se ha definido *sistema (informático) consciente del contexto* (i.e., contextualizado) como sigue (Dey 2001):

*A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task.*

Adomavicius et al. (2011) caracterizaron *contexto* dentro de los sistemas de recomendación teniendo en cuenta dos aspectos: por un lado, la cantidad de información contextual que un sistema conoce de manera explícita, y por otro lado, cómo esa información contextual cambia con el tiempo. Así, dependiendo de lo que el sistema de recomendación conoce acerca del contexto, los autores distinguen tres **tipos de contexto**:

- **Contexto completamente observable (fully observable):** el sistema conoce de forma explícita la información contextual, tanto su estructura como sus valores. Por ejemplo, el sistema conoce que existe información temporal, y además conoce que está estructurada en tres tipos: días laborables, fines de semana y vacaciones.
- **Contexto parcialmente observable (partially observable):** el sistema sólo conoce de forma explícita una parte de la información contextual, por ejemplo, conoce la existencia de información temporal, pero desconoce cómo está estructurada internamente.
- **Contexto no observable (unobservable):** el sistema no tiene disponible de forma explícita información alguna acerca del contexto; ésta se encuentra implícita o latente, y el sistema ha de extraerla o inferirla e.g., usando modelos predictivos.

Por otro lado, dependiendo de si la información contextual cambia o no con el tiempo, Adomavicius et al. (2011) distinguen los siguientes dos escenarios de **dinámica de contexto**:

- **Contexto estático:** la información contextual y su estructura interna permanecen fijas en el tiempo.
- **Contexto dinámico:** la información contextual cambia de alguna manera, por ejemplo, ciertos factores dejan de ser relevantes con el tiempo, o la estructura interna de contexto cambia apareciendo nuevas categorías y valores contextuales.

Combinado los dos aspectos anteriores, se obtiene el diagrama que se muestra en la Figura 2.1, que permite clasificar los **sistemas de recomendación conscientes del contexto (CARS)**:

How Contextual Factors Change	Knowledge of the RS about the Contextual Factors		
	Fully Observable	Partially Observable	Unobservable
Static	Everything Known about Context	Partial and Static Context Knowledge	Latent Knowledge of Context
Dynamic	Context Relevance Is Dynamic	Partial and Dynamic Context Knowledge	Nothing Is Known about Context

**Figura 2.1** Clasificación de CARS según Adomavicius et al. (2011)

La mayor parte de los trabajos existentes acerca de sistemas de recomendación conscientes del contexto se encuadran dentro del ámbito estático y completamente observable, es decir, donde el sistema conoce absolutamente todo acerca del contexto,

tanto los factores contextuales que presentan como su estructura interna, y además la información contextual permanece fija durante el tiempo.

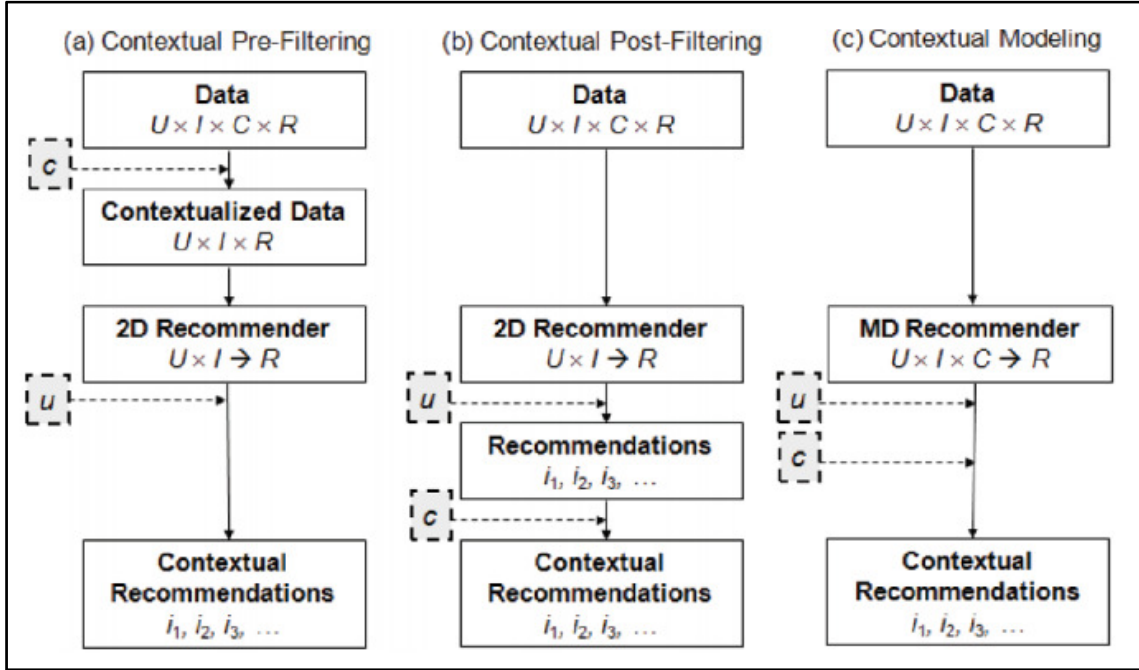
Como se verá en capítulos posteriores, la aproximación presentada en este trabajo podría encuadrarse dentro de los ámbitos *no observable* (y *estático*), ya que se desconocen de forma explícita tanto los aspectos y valores contextuales que se tratan en *reviews* textuales de los usuarios, como las categorías y estructura contextuales a usar en el sistema. Para abordar estas situaciones, se extraerá la información de contexto latente en las *reviews*, y se asociará a una taxonomía de contexto generada semi-automática a partir de la base de conocimiento DBpedia.

Por otra parte, desde el punto de vista algorítmico, se pueden distinguir tres **estrategias para incorporar y hacer uso de la información contextual en la generación de recomendaciones**. En la Figura 2.2, obtenida de (Adomavicius & Tuzhilin, 2011), se presentan las tres formas en las que se puede tratar dicha información:

- **Pre-filtrado contextual (contextual pre-filtering)**: desde esta perspectiva la información contextual se emplea para seleccionar interacciones usuario-ítem relevantes al contexto actual, y con ellas generar las recomendaciones empleando cualquiera de los algoritmos de recomendación tradicionales para las dos dimensiones  $U \times I$ . El contexto  $C$  por tanto sirve para filtrar los datos originales. Esto puede suponer un problema, pues el conjunto de datos resultante podría ser demasiado reducido. Para abordar este problema, se suelen reemplazar los contextos específicos a otros más generales, por ejemplo ‘sábado’  $\rightarrow$  ‘fin de semana’. En principio, esta aproximación facilitaría el generar mejores recomendaciones que empleando todos los datos ignorando el contexto, ya que a pesar de que se utilicen menos datos, estos se entienden como más relevantes y menos ruidosos. No obstante, esto puede depender del dominio de aplicación, así como de las características de los datos disponibles.
- **Post-filtrado contextual (contextual post-filtering)**: desde esta perspectiva, en un principio se ignora la información contextual y se generan las recomendaciones utilizando un algoritmo tradicional de dos dimensiones  $U \times I$  con todos los datos. Después, la información contextual se emplea para reducir o reordenar esas recomendaciones, filtrando o bajando en el *ranking* aquellas recomendaciones que son irrelevantes dentro del contexto actual del usuario.
- **Modelado contextual (contextual modeling)**: desde esta perspectiva la información contextual se emplea directamente por los algoritmos de recomendación. Estos algoritmos se suelen basar en aproximaciones de recomendación tradicionales de dos dimensiones  $U \times I$ , pero extendiéndose para explotar datos multi-dimensionalidad  $U \times I \times C$ , y poder tratar las información



contextual. Ejemplos típicos de estos algoritmos son los métodos basados en factorización de matrices (Baltrunas et al., 2011).



**Figura 2.2** Estrategias de incorporación y uso de información contextual en CARS según Adomavicius y Tuzhilin (2011)

### 2.1.5 Evaluación de los sistemas de recomendación

En la última década ha habido un gran número de trabajos de investigación en el campo de los sistemas de recomendación, donde la mayoría han estado centrados en diseñar nuevos algoritmos, de manera que estos eran comparados con algoritmos de recomendación existentes mediante alguna métrica de evaluación.

Inicialmente y como indican Gunawardana y Shani (2011), la mayoría de estos sistemas eran evaluados revisando la *precisión* del sistema (la habilidad de predecir los gustos del usuario de manera precisa). Sin embargo, esto no es del todo correcto, ya que los usuarios no siempre emplean los sistemas de recomendación para encontrar ítems que representen exactamente sus gustos, sino que también buscan descubrir nuevos ítems, obtener ítems más diversos, que el sistema funcione rápidamente... Por esta razón, las investigaciones han ido centrándose cada vez más en aquellas métricas que se encargan de evaluar el ranking de recomendaciones más que en aquellas orientadas a la predicción del valor de rating de los diferentes ítems.

Antes de continuar hablando de las métricas de evaluación de predicción de rating, y de evaluación del ranking de recomendaciones, es necesario comentar los distintos tipos de experimentos que se pueden llevar a cabo para evaluar los sistemas de recomendación (Gunawardana y Shani, 2011):

- **Experimentos off-line:** es el método más empleado para evaluar sistemas de recomendación ya que es el más sencillo de llevar a cabo, y permite poder evaluar un gran número de algoritmos de recomendación con un coste bajo. Consiste en emplear un conjunto de datos de usuarios, ítems y sus ratings, que haya sido recogido en algún momento previo al momento de la evaluación a partir de usuarios, que se asume tendrán el mismo comportamiento que los usuarios finales del sistema de recomendación. La parte buena de este método es que no se requiere interacción con el usuario durante el proceso de evaluación; sin embargo, es necesario asumir que los usuarios que interactuaron con el sistema en el momento de recoger las evaluaciones, van a tener un comportamiento similar a aquellos usuarios que empleen el sistema cuando éste se encuentre en un entorno de producción, además de que en el momento de realizar la evaluación, no se puede disponer de información adicional como por ejemplo la experiencia del usuario o su opinión durante su interacción con el sistema. Este tipo de experimentos se emplean también para filtrar aquellos algoritmos de recomendación inapropiados, para después poder emplear otras técnicas de evaluación más costosas (i.e., estudios de usuario y evaluaciones on-line) sobre un número de algoritmos más acotado.
- **Estudios de usuario:** los estudios de usuario no sólo recogen la información directa que el usuario proporciona al sistema (e.g., ratings), sino que también recogen las interacciones de los usuarios con el sistema. Estos estudios consisten en monitorizar y observar a un grupo de usuarios mientras realizan diversas tareas con el sistema real recopilando información cuantitativa (e.g., el número de tareas completadas, el tiempo empleado en cada tarea...) y cualitativa (e.g., la opinión acerca de la interfaz de usuario, si las tareas le han parecido sencillas...). En ellos, la información obtenida para llevar a cabo la evaluación del sistema es mucho mayor que en los métodos anteriores; no obstante, los estudios de usuario son más costosos de llevar a cabo, tanto en tiempo como económicamente (si se realiza algún tipo de compensación económica por participar en el estudio)
- **Evaluación on-line:** este tipo de evaluación permite comprobar el efecto del sistema de recomendación en usuarios reales que se encuentran interactuando con el sistema para realizar tareas reales. Por esta razón, muchos sistemas que ya se encuentran en producción emplean este tipo de evaluación para poder comparar múltiples soluciones de manera real. Una de las técnicas más utilizadas es la denominada Test A/B, que consiste en redirigir una pequeña parte del tráfico del sistema hacia un sistema de recomendación alternativo y guardar esas interacciones de los usuarios con ese nuevo sistema. Es importante que esos usuarios que se redirigen, se escojan de manera aleatoria y que en ningún momento sepan que están utilizando un sistema diferente al esperado, ya

que podría influenciar su interacción, para lo cual es importante mantener fija la interfaz de usuario. Por otro lado, este tipo de evaluación tiene alguna desventaja, como puede ser que si el sistema que se está evaluando frente al sistema real proporciona resultados irrelevantes, se corre el riesgo de que los usuarios abandonen el sistema real y no vuelvan a utilizarlo, lo que puede suponer un fracaso si se trata de una aplicación comercial. Por esta razón, estas evaluaciones on-line deben ser empleadas al final del proceso de evaluación, tras haber llevado a cabo experimentos off-line e incluso estudios de usuario.

Dentro de los experimentos off-line, como se ha comentado previamente, las métricas de evaluación pueden clasificarse en dos tipos:

- **Métricas de predicción de rating de ítems:** se basan en evaluar la precisión con la que los ratings son estimados por el sistema, por tanto miden la diferencia entre el valor predicho de utilidad (normalmente el *rating*) y el valor real de utilidad, siendo los métodos estadísticos más utilizados, el cálculo del error medio absoluto (*Mean Absolute Error* o MAE), y la raíz cuadrada del error cuadrático medio (*Root Mean Squared Error* o RMSE).

$$MAE = \frac{\sum_{r_{u,i} \in \text{Conjunto Test}} |\hat{r}_{u,i} - r_{u,i}|}{|\text{Conjunto Test}|}$$

$$RMSE = \sqrt{\frac{\sum_{r_{u,i} \in \text{Conjunto Test}} (\hat{r}_{u,i} - r_{u,i})^2}{|\text{Conjunto Test}|}}$$

En ambos casos, cuando menor sea el valor (tanto MAE, como RMSE), significa que se obtiene una mayor precisión del sistema de recomendación. No obstante, este tipo de métricas han empezado a caer en el desuso en favor de aquellas métricas que permiten evaluar ranking de recomendaciones.

- **Métricas de evaluación de ranking de recomendaciones:** analizan el orden/posición de los ítems relevantes dentro del listado propuesto por el sistema. Algunas de estas métricas son *Precision* a nivel k,  $P@k$ , mide la fracción de ítems de los 'k' primeros del ranking que son relevantes para el usuario; o *Recall* a nivel k,  $R@k$ , que mide la fracción de ítems relevantes para el usuario que fueron recomendados entre los 'k' primeros elementos del ranking.

$$P@k = \frac{1}{|U|} \sum_{u \in U} \frac{|Rel_u@k|}{k}$$

$$R@k = \frac{1}{|U|} \sum_{u \in U} \frac{|Rel_u@k|}{|Rel_u|}$$

donde  $Rel_u$  representa el conjunto de ítems relevantes para el usuario 'u',  $Rel_u@k$  representa el conjunto de ítems relevantes para el usuario 'u' que se

encuentran dentro de los 'k' primeros ítems devueltos por el sistema de recomendación.

Otra métrica de evaluación de ranking empleada habitualmente es MAP (*Mean Average Precision*), que tiene en cuenta el orden relativo de los ítems relevantes dentro del ranking recomendado, calculando la *Precision* en aquellas posiciones del ranking donde se encuentra un ítem relevante para el usuario.

$$AP(u)@k = \frac{1}{|Rel_u|} \sum_{n=1}^k P@n \cdot 1(i_n \in Rel_u)$$

$$MAP = \frac{1}{|U|} \sum_{u \in U} AP(u)$$

donde  $P@k$  representa la precisión a nivel k explicada anteriormente. Por último, la métrica nDCG (*normalized Discounted Cumulative Gain*) tiene en cuenta los distintos niveles de relevancia que pueden existir para obtener los ítems más relevantes, de esta forma, si los ítems más relevantes son devueltos en posiciones más elevadas del ranking esto contribuye a que el sistema sea de calidad.

$$DCG_u@k = \sum_{n=1}^k \frac{2^{rel_u(i_n)} - 1}{\log(n + 1)}$$

$$nDCG@k = \frac{1}{|U|} \sum_{u \in U} \frac{DCG_u@k}{IDCG_u@k}$$

donde  $rel_u(i_n)$  mide el grado de relevancia del ítem para el usuario u cuando se ha obtenido en la posición n del ranking, y  $IDCG_u@k$  representa DCG@k para el ranking ideal del usuario u, es decir, donde cada ítem relevante es devuelto en la posición idónea del ranking (el más relevante primero, después el segundo más relevante, ...).

Además de estas métricas, existen otras propiedades que se suelen medir para evaluar los sistemas de recomendación, como pueden ser:

- **Cobertura.** Puede darse el caso de que se tenga un sistema con el que se obtienen muy buenos resultados sólo para una pequeña porción de los ítems donde se tienen datos suficientes, mientras que, para el resto, no se tenga una precisión acertada. El termino cobertura suele referirse a la proporción de ítems que el sistema es capaz de recomendar de entre todos los que tiene disponible (habitualmente conocido como cobertura de catálogo), no obstante, en otras ocasiones también puede referirse a la proporción de usuarios para los cuales se pueden proporcionar recomendaciones, ya que a veces el sistema puede no dar ninguna recomendación a determinados usuarios.

- **Confianza.** La confianza de un sistema de recomendación puede definirse como la credibilidad que tiene el sistema en sus predicciones y recomendaciones. Esta propiedad es importante ya que si el usuario obtiene una recomendación donde el sistema tiene un alto nivel de confianza, es posible que el usuario consuma el ítem recomendado, mientras que si el nivel de confianza es bajo, lo más probable es que el usuario termine buscando información adicional para decidir si consume o no el ítem.
- **Novedad.** Dentro de los sistemas de recomendación, la novedad se centra en aquellos ítems recomendados que el usuario desconocía. Para tenerla en cuenta bastaría con hacer un filtrado de aquellos ítems recomendados que el usuario ya conoce. Sin embargo, esto no siempre es sencillo, ya que muchas veces el usuario no da información completa de todos aquellos ítems que ya ha consumido. Por otro lado, cuando se quieren recomendar ítems basados en su novedad, es importante controlar la precisión del sistema, ya que, si las recomendaciones obtenidas son novedosas, pero irrelevantes para el usuario, el proceso de recomendación en sí mismo será también irrelevante. Por eso, una aproximación puede consistir en considerar y buscar la novedad sólo entre aquellos ítems que son relevantes.
- **Diversidad.** Se puede definir diversidad como lo contrario de similitud. Un buen sistema de recomendación debe proporcionar resultados diversos al usuario, ya que si los ítems recomendados son demasiados similares entre sí, puede que en su conjunto no sean de utilidad para el usuario. Por ejemplo, si se trata de un sistema de recomendación de lugares vacacionales, presentar todas las recomendaciones sobre la misma localización variando únicamente el hotel, o la actividad principal a realizar, puede ser menos útil que si se hubiesen recomendado distintas localizaciones que el usuario pueda explorar.
- **Serendipia.** Se corresponde con la sensación que se produce cuando ocurre un hallazgo afortunado e inesperado al estar buscando otra cosa distinta. Por ejemplo, si un usuario valoró positivamente películas de un actor en concreto, obtener una recomendación de otra película de ese mismo actor puede ser algo novedoso (si el usuario no la ha visto), pero no sorprendente; sin embargo, si obtiene una recomendación de otra película con un reparto totalmente diferente y que además es de su agrado, se puede producir una sensación de serendipia, ya que el usuario no esperaría haber obtenido esa película recomendada. También se puede pensar la serendipia como la cantidad de información relevante que es nueva para el usuario a la hora de obtener la recomendación.

## 2.2 Web Semántica

La Web Semántica es una Web extendida dotada de mayor significado que la Web tradicional. Se basa en la idea de añadir metadatos semánticos y ontológicos a la World Wide Web (WWW) para describir el contenido, el significado y la relación de los datos que se encuentran en la Web, mediante lenguajes formales, para que puedan ser tratados de manera automática.

Al trabajar con la Web Semántica se utilizan tecnologías basadas en el estándar RDF, empleado para describir los distintos recursos de la Web y las relaciones que puedan existir entre ellos; en OWL, un lenguaje empleado para definir ontologías de manera estructurada para que puedan ser utilizadas a través de distintos sistemas, realmente es una extensión de RFD aunque más completo; y SPARQL, un lenguaje de consulta sobre RDF que permite hacer búsquedas sobre los recursos de la Web anotados semánticamente.

### 2.2.1 Ontologías y la Web Semántica

La Web Semántica propone dotar a la Web de un mayor significado y una mayor estructura global de sus contenidos, clasificándolos y anotándolos con semántica explícita que sea procesable por las máquinas (Castells 2003). De esta forma, si se piensa en la Web como un grafo, la Web actual estaría formada por nodos del mismo tipo (documentos HTML), cuyos arcos (hiperenlaces) son de semántica indistinguible. Así, bajo esa representación no se podría distinguir una página de otra totalmente distinta, como tampoco se diferenciaría de manera explícita los distintos tipos de enlaces. Por el contrario, en la Web Semántica, cada nodo (llamado recurso) es de un determinado tipo, y los arcos representan relaciones explícitas y diferenciadas entre los distintos recursos, como se puede ver en la siguiente figura:

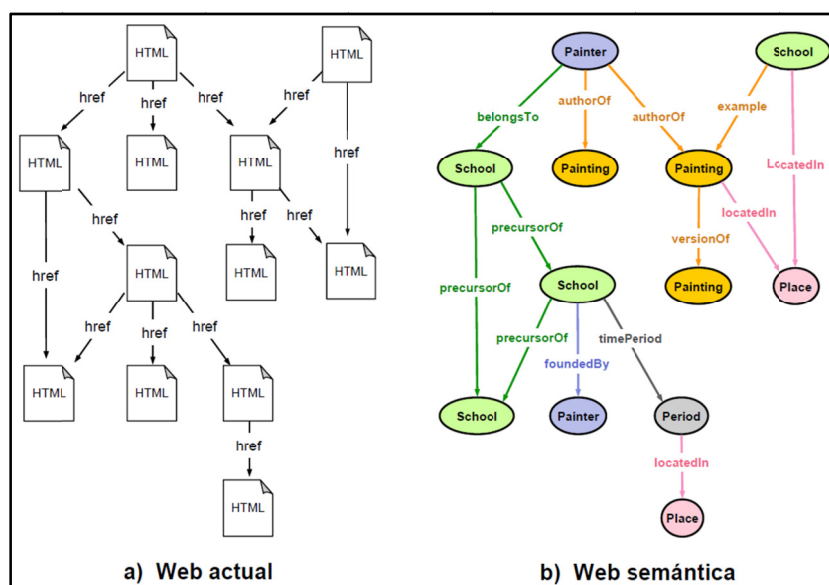


Figura 2.3 Web actual contra Web Semántica (Castells, 2003)

En ella se observa que en la Web Semántica se puede diferenciar bien los distintos tipos de recursos (e.g., pintor, pintura, escuela artística, ...) y cómo están relacionados entre sí (e.g., pintor-obra, pintor-escuela, ...).

Para llevar a cabo esta transformación, es necesario utilizar un marco de trabajo común que permita crear estos nuevos conceptos, relacionarlos entre sí y poder ser comprendidos tanto por usuarios como por máquinas. Todo esto se consigue mediante el uso de ontologías.

Una ontología puede definirse como un conocimiento compartido entre varios dominios que normalmente se expresa mediante entidades, relaciones, funciones, axiomas e instancias. Son modelos que tienen un alto grado de normalización y formalización, por lo que están bastante extendidos. En la imagen anterior, se puede observar las distintas entidades existentes tales como ‘pintor’, ‘pintura’ o ‘lugar’ y la relación existente entre dichas entidades, por ejemplo, pintura ‘perteneiente’ a un pintor concreto o pintura ‘localizada’ en un lugar en concreto.

Para definir las ontologías de manera formal y que sean comprensibles tanto por las personas como por las máquinas es necesario utilizar un lenguaje, ahí es donde entra RDF (*Resource Description Framework*), que a día de hoy es el estándar de W3C para la Web Semántica. RDF permite describir los distintos recursos indicando las propiedades que tienen asignadas, así como sus valores. Para ello, su elemento básico es la ‘tripleta’ [sujeto, predicado, objeto], que permite vincular dos recursos –sujeto y objeto– mediante una propiedad (predicado) que los une.

En la siguiente figura se puede observar un extracto de representación RDF, con varias tripletas. En ellas, el ‘sujeto’ está representado por la URI correspondiente a la página de Iván Cantador, profesor de la Universidad Autónoma de Madrid (UAM); los ‘objetos’ son los valores de las distintas propiedades existentes como son el nombre, el apellido o la nacionalidad, que no son otra cosa que los ‘predicados’ de las tripletas. Como puede observarse, la sintaxis de RDF está basada en XML (*eXtensible Markup Language*).

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:epsp="http://www.eps.uam.es/personnel#"
  <rdf:Description
    rdf:about="http://www.eps.uam.es/personnel/IvanCantador">
      <epsp:name>Ivan</epsp:name>
      <epsp:lastname>Cantador</epsp:lastname>
      <epsp:nationality>Spanish</epsp:nationality>
      <rdf:type>Assistant Professor</rdf:type>
    </rdf:Description>
  </rdf:RDF>
```

**Figura 2.4** Ejemplo de representación RDF para el profesor Iván Cantador

Como extensión a RDF, RDF Schema (RDFS) proporciona el *framework* adecuado para que se puedan describir clases y propiedades, así como las relaciones existentes entre ellas para poder crear jerarquías. En la siguiente figura se puede ver un ejemplo de la representación RDFS (también basada en XML). En ella se observan las distintas ‘clases’ que existen como ‘profesor’ y ‘profesor asistente’ y la ‘jerarquía’ que hay entre ellas, como puede ser que ‘profesor ayudante’ represente una subclase de ‘profesor’.

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://www.eps.uam.es#
  <rdfs:Class rdf:ID="Professor" />
  <rdfs:Class rdf:ID="Assistant Professor">
    <rdfs:subClassOf rdf:resource="#Professor"/>
  </rdfs:Class>
  <rdf:Property rdf:ID="teachesSubject">
    <rdfs:Domain rdf:resource="#Professor"/>
    <rdfs:Range rdf:resource="#Subject"/>
  </rdf:Property>
</rdf:RDF>
```

**Figura 2.5** Ejemplo de representación RDFS

Más adelante, se definió OWL (*Web Ontology Language*), construido también sobre RDF (por lo que puede considerarse una extensión del mismo) y que por tanto también emplea representación sintáctica XML. La ventaja de OWL es que permite definir ciertas propiedades en las relaciones, como puede ser la cardinalidad, y restricciones para que un objeto pueda pertenecer a una determinada clase (e.g., para pertenecer a la clase ‘profesor’ es necesario contar con un ‘título académico’ con valor equivalente al ‘doctorado’). Esto a su vez da la posibilidad de utilizar expresiones lógicas para determinar por ejemplo si un determinado objeto puede ser ‘profesor’, o para combinar distintas clases con operadores lógicos, para así crear otras clases nuevas. En la siguiente figura puede observarse un extracto de la representación OWL donde se define la clase ‘profesor’ y la restricción mencionada anteriormente.



```

<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.eps.uam.es#"

  <owl:Class rdf:ID="Professor">
    <owl:restriction>
      <owl:onProperty rdf:resource="#hasAcademicTitle"/>
      <owl:hasValue>
        PhD^^http://www.w3.org/2001/XMLSchema#string</owl:hasValue>
      <owl:restriction>
    </owl:Class>
  </rdf:RDF>

```

Figura 2.6 Ejemplo de representación OWL

Para acceder a una base de datos en RDF existe, entre otros, el lenguaje SPARQL (*SPARQL Protocol and RDF Query Language*), el cual permite construir consultas sobre esos datos para extraerlos. Al acceder sobre datos RDF, también se basa en la representación de la ‘tripleta’ [sujeto, predicado, objeto]. En la siguiente figura se puede ver un ejemplo de una consulta SPARQL para extraer los nombres de todos aquellos objetos que sean del tipo ‘profesor ayudante’ y que tengan la nacionalidad ‘española’.

```

SELECT ?x ?name
WHERE
{
  ?x rdf:type epsp:Assistant Professor.
  ?x epsp:name ?name.
  ?x epsp:nationality "Spanish"}

```

Figura 2.7 Ejemplo de consulta SPARQL

SPARQL permite construir consultas de lo más complejas utilizando condiciones para filtrar elementos, ordenar los resultados, obtener hasta un determinado número de resultados, realizar operaciones de agrupación como contar, calcular máximos/mínimos, calcular la media, etc.

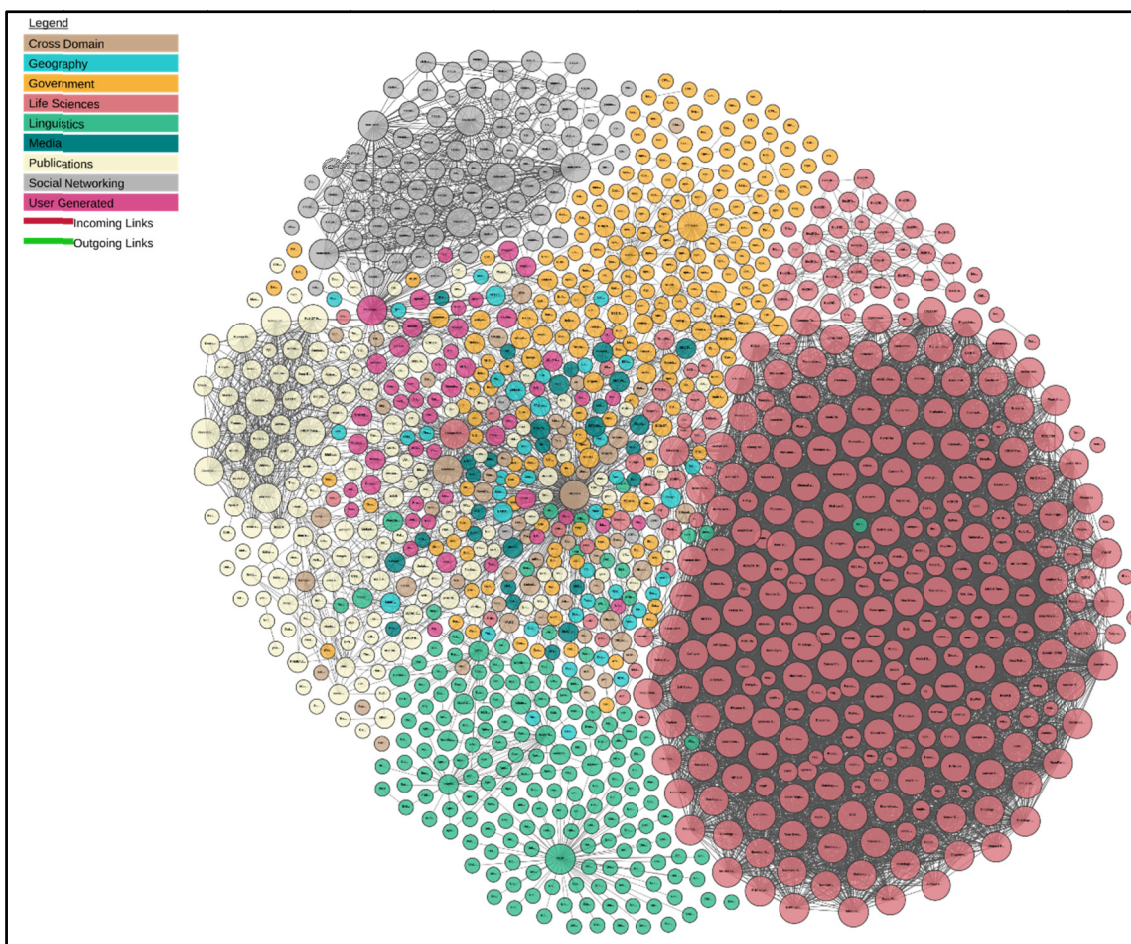
### 2.2.2 El proyecto Linked Open Data

Antes de explicar el proyecto Linked Open Data, es necesario comentar que es el concepto de Open Data, (Datos Abiertos). Grosso modo, no es otra cosa que una práctica que tiene como objetivo que los datos se encuentren a libre disposición para las personas e instituciones.

Linked Open Data es un proyecto desarrollado por el grupo W3C encargado de estandarizar y divulgar la Web Semántica, donde distintos datos abiertos se encuentran descritos en estándares de la Web Semántica (como RDF) y están enlazados entre sí,

de forma que el usuario puede obtener datos que provienen de distintas fuentes, instituciones y organizaciones, para así poder explorar y combinar esos datos de manera libre y sin restricciones.

En definitiva, se puede entender Linked Open Data como un gigantesco grafo de recursos (datos estructurados) interconectados entre sí, de forma que cada recurso está descrito mediante RFD, con sus características y enlaces a otros recursos relacionados, o bien que representan ese mismo recurso dentro de *datasets* diferentes. En la siguiente figura se muestra el conjunto de los distintos datasets enlazados dentro del proyecto Linked Open Data (representados mediante nodos de un grafo), así como el dominio al que pertenecen (indicados mediante los distintos colores de los nodos).



**Figura 2.8** Linking Open Data cloud diagram 2017-02-20. <http://lod-cloud.net/>

Para referenciar y acceder a un recurso se utiliza su URI (*Uniform Resource Identifiers*), con la que queda se identificado de forma única, y se vincula a otros recursos indicando sus URIs, lo cual crea una red de datos interconectados entre sí, que pueden ser explorados tanto por personas como por máquinas. No obstante, para hacer uso de esta información se requieren ciertas acciones como por ejemplo configurar los servidores web para que devuelvan estos conceptos en lugar de documentos, o utilizar

buscadores avanzados que estén preparados para trabajar con esos datos enlazados y descubrir las relaciones entre los distintos datasets.

### 2.2.3 DBpedia

El proyecto DBpedia es un proyecto encargado de extraer datos de Wikipedia (Figura 2.9) y componer una versión semántica/estructurada de los mismos, empleando RDF como formato de almacenamiento, de forma que puedan consultarse y recuperarse mediante consultas a bases de datos empleando lenguaje como SPARQL.



Figura 2.9 Página web de Wikipedia de la Universidad Autónoma de Madrid<sup>6</sup>

Al contrario que en Wikipedia, donde toda la información está almacenada como documentos, en DBpedia, la información extraída se almacena de manera estructurada, lo que permite trabajar con los datos de manera más flexible y compleja, pudiendo realizar consultas para extraer por ejemplo todo lo que ocurrió en un determinado periodo de tiempo, los recursos que tienen una característica dada, o elementos que están relacionados mediante cierta propiedad.

La Figura 2.10 muestra la página DBpedia para la Universidad Autónoma de Madrid. Como se puede observar, representa información que la que se encuentra dentro de Wikipedia (Figura 2.9), pero de una manera estructurada:

<sup>6</sup> UAM en Wikipedia, [https://es.wikipedia.org/wiki/Universidad\\_Autónoma\\_de\\_Madrid](https://es.wikipedia.org/wiki/Universidad_Autónoma_de_Madrid)

**About: Universidad Autónoma de Madrid**  
 An Entity of Type : [Organisation](#), from Named Graph : <http://es.dbpedia.org>, within Data Space : [es.dbpedia.org](#)

La Universidad Autónoma de Madrid (UAM) es una universidad pública que se fundó en 1968, momento en que sus facultades estaban dispersas por diversos edificios de la capital española. No obstante, la localización actual de esta universidad es el campus de Cantoblanco, al norte de la ciudad de Madrid, junto a Alcobendas y San Sebastián de los Reyes. Dicho campus, con 2.252.000 m² de superficie total, se inauguró el 25 de octubre de 1971.

Property	Value
dbpedia-owl:abstract	La Universidad Autónoma de Madrid (UAM) es una universidad pública que se fundó en 1968, momento en que sus facultades estaban dispersas por diversos edificios de la capital española. No obstante, la localización actual de esta universidad es el campus de Cantoblanco, al norte de la ciudad de Madrid, junto a Alcobendas y San Sebastián de los Reyes. Dicho campus, con 2.252.000 m² de superficie total, se inauguró el 25 de octubre de 1971. Debido a las características del tardofranquismo atrajo a profesores más jóvenes, varios de ellos serían depurados políticamente, que marcaron una impronta muy distinta al espíritu de la universidad. Es una de las seis universidades públicas de la Comunidad de Madrid junto a la Universidad Complutense de Madrid, la Universidad Carlos III de Madrid, la Universidad Politécnica de Madrid, la Universidad de Alcalá y la Universidad Rey Juan Carlos. Cuenta con siete facultades: Ciencias, Derecho, Filosofía y Letras, Psicología, Medicina (situada fuera del Campus de Cantoblanco), Ciencias Económicas y Empresariales, Formación de Profesorado y Educación y la Escuela Politécnica Superior, además de cuatro Escuelas Universitarias adscritas, todo ello estructurado en 70 Departamentos. También cuenta con numerosos Institutos de investigación propios y centros del Consejo Superior de Investigaciones Científicas (CSIC) asociados. Destacan sus estudios de Medicina, gracias a su prestigioso Hospital Universitario La Paz, en Biología, por su estrecha relación con diversos centros del CSIC y los discípulos de Severo Ochoa; en matemáticas, ya que suele figurar entre los cincuenta mejores del mundo en los estudios internacionales más reconocidos; de Historia, estudios puestos en marcha por el académico, y Premio Príncipe de Asturias, Miguel Artola; y de Derecho, situados como los mejores de España en diversos rankings, por la importancia de sus profesores, entre los que se contaba Francisco Tomás y Valiente.
dbpedia-owl:chancellor	dbpedia:José_María_Sanz_Martínez
dbpedia-owl:city	dbpedia:Madrid
dbpedia-owl:country	dbpedia:España
dbpedia-owl:motto	Quid Ultra Faciam? ¿Qué más debemos hacer? Quid Ultra Faciam? ¿Qué más debemos hacer?
dbpedia-owl:numberOfStudents	32446 (xsd:integer)
dbpedia-owl:provost	dbpedia:José_María_Sanz_Martínez
dbpedia-owl:thumbnail	<a href="http://commons.wikimedia.org/wiki/Special:FilePath/Escudo_Autónoma.PNG?width=300">http://commons.wikimedia.org/wiki/Special:FilePath/Escudo_Autónoma.PNG?width=300</a>
dbpedia-owl:wikiPageExternalLink	<a href="http://www.iulce.es">http://www.iulce.es</a> <a href="http://www.uam.es">http://www.uam.es</a>
dbpedia-owl:wikiPageID	62688 (xsd:integer)
dbpedia-owl:wikiPageLength	33405 (xsd:integer)
dbpedia-owl:wikiPageOutDegree	280 (xsd:integer)
dbpedia-owl:wikiPageRevisionID	74057050 (xsd:integer)
dbpedia-owl:wikiPageWikiLink	dbpedia:ONCE dbpedia:Cayetano_López_Martínez

**Figura 2.10** Página web en DBpedia para la Universidad Autónoma de Madrid<sup>7</sup>

Como se puede observar, en DBpedia la forma de estructurar la información de un recurso es mediante pares propiedad-valor. En el ejemplo de la Universidad Autónoma de Madrid, se observa como existen propiedades para representar la ‘ciudad’, el ‘país’ o el ‘rector’ de la universidad; información se encuentra de manera no estructurada en Wikipedia.

La versión inglesa de DBpedia cuenta con 4.58 millones de elementos, de los cuales 4.22 son representados de manera consistente mediante ontologías, incluyendo un total de 1.445.000 personas, 735.000 lugares, 411.000 trabajos creativos (incluyendo 123.000 álbumes musicales, 87.000 películas y 19.000 videojuegos), 241.000 organizaciones (incluyendo 58.000 compañías y 49.000 instituciones educativas), 251.000 especies y 6.000 enfermedades. A su vez, existen diferentes versiones de DBpedia según el idioma utilizado, contando en total con 125 idiomas, lo que a su vez representa un total de 38.3 millones de elementos contando los distintos elementos representados en todos los idiomas y almacenados en DBpedia.

La información de DBpedia está accesible y disponible bajo los términos de la licencia Creative Commons Attribution-ShareAlike 3.0 (CC BY-SA 3.0) y la licencia GNU Free Documentation.

<sup>7</sup> UAM en DBpedia, [http://es.dbpedia.org/page/Universidad\\_Autónoma\\_de\\_Madrid](http://es.dbpedia.org/page/Universidad_Autónoma_de_Madrid)



---

Para concluir la sección, comentar que existen multitud de herramientas y aplicaciones que trabajan sobre DBpedia. Por ejemplo, DBpedia Spotlight es una herramienta que genera anotaciones de un texto con recursos de DBpedia y que, por tanto, representa un nexo para enlazar información textual desestructurada con información estructurada del proyecto Linked Open Data, a través de DBpedia. Así, esta herramienta reconoce los distintos objetos que aparecen en el texto y los enlaza con sus identificadores dentro de DBpedia, para poder emplearlos posteriormente en distintas tareas como pueden ser la extracción y detección de entidades, y la clasificación de documentos por temática.



## Capítulo 3

# Trabajo relacionado

La propuesta presentada en este trabajo está compuesta de tres componentes principales, que son temas y líneas de investigación en diferentes áreas. La primera de esas componentes trata el *modelado de contexto*, destinado a la representación y almacenamiento de información contextual genérica. La segunda se corresponde con la *anotación semántica de textos*; en concreto con la identificación de aspectos y valores contextuales en reviews (evaluaciones escritas) de usuarios sobre ítems. Las reviews han de ser procesadas adecuadamente para extraer de ellas la información contextual latente en las valoraciones de los usuarios, y para asociarla a cierto modelo de contexto desarrollado. La tercera y última componente es la asociada a la *recomendación contextualizada*, que consiste en la adaptación y uso de algoritmos de recomendación conscientes de contexto explotando la información contextual obtenida de las reviews.

Para cada una de estas tres componentes se ha hecho un estudio previo de la literatura existente con el fin de identificar trabajo relacionado que sirva como punto de partida para conseguir los objetivos propuestos en el trabajo, y a su vez permita determinar qué puede ofrecer este trabajo frente a los ya publicados. En este capítulo se presenta el estudio del arte realizado sobre las tres componentes citadas: modelado de contexto, reconocimiento de información contextual en texto, y recomendación contextualizada.

### 3.1 Modelado de contexto

Los sistemas informáticos conscientes del contexto ofrecen nuevas oportunidades tanto para los desarrolladores como para los investigadores. En relación a estos últimos, cada vez se produce un mayor número de artículos con respecto a este tema, por lo que representan un área de investigación prometedora. Por tanto, son numerosos los trabajos que se han encargado de recopilar información acerca de los sistemas

informáticos conscientes del contexto, así como del modelado de contexto en si mismo. Chen y Kotz (2000) realizaron un estudio de distintas aplicaciones conscientes del contexto en el ámbito de la computación móvil, incluyendo además distintas aproximaciones para capturar y modelar el contexto. Más tarde, Baldauf et al. (2007) realizaron un estudio más extenso acerca de este tipo de sistemas, describiendo una serie de principios comunes en la arquitectura de este tipo de sistemas empleando un sistema de capas (aplicación, almacenamiento, sensores, etc.). Posteriormente, Hong et al. (2009) realizaron un estudio bastante extenso acerca de la literatura existente en este campo entre los años 2000 y 2007, clasificando los distintos trabajos en cinco grandes grupos según el tema que tratan: conceptos e investigación, comunicaciones, middleware, aplicaciones e infraestructura de usuario.

Para el caso concreto de este trabajo, el contexto se entiende como aquellos aspectos contextuales representativos extraídos de evaluaciones textuales dadas por usuarios al valorar productos de diversa índole, como películas, música y libros. Por tanto, representan los aspectos bajo los cuales esos productos fueron utilizados o consumidos por los usuarios. Como se ha dicho anteriormente, dicha información contextual se usará para realizar recomendaciones personalizadas más efectivas.

Una vez definido el contexto dentro de un sistema o aplicación, es necesario llevar a cabo un modelado del mismo. El modelado o representación del contexto sirve por un lado para identificar qué categorías y valores de contexto son susceptibles de emplearse dentro del sistema, y por otro lado para almacenar la información contextual asociada al conjunto de los datos del sistema.

En la literatura existen números trabajos que analizan distintas formas de modelar el contexto dentro de sistemas informáticos, desde un enfoque genérico para que puedan utilizarse independientemente de cual sea el dominio de aplicación. Abowd et al. (1999) identificaron que ciertos tipos o categorías de contexto son más importantes que otros, ya que sirven para identificar mejor las situaciones en las que se encuentra un usuario. Estos contextos principales, denominados de primer nivel, son la **identidad** (quien es el usuario), la **ubicación** (dónde está el usuario), la **actividad** (qué está haciendo el usuario) y el **tiempo** (cuando lo está haciendo). A partir de estos contextos de primer nivel pueden extenderse distintos subniveles que especifiquen o especialicen el contexto del usuario, e.g., un número de teléfono podría usarse como contexto secundario del contexto identidad.

En el trabajo realizado por Strang y Linnhoff-Popien (2004) se lleva a cabo un análisis de distintas formas de modelar el contexto dentro del ámbito de la computación ubicua, atendiendo al modo en el que se estructuran los datos. Los modelos tratados incluyen estructuras de pares ‘clave-valor’, modelos orientados a objetos, basados en lógica y basados en ontologías. La idea es que estos modelos sean lo más genéricos posibles para así permitir la interoperabilidad de aplicaciones de distintos dominios. Como dicho



estudio se realiza dentro del ámbito de la Computación Ubicua (o Inteligencia Ambiental), se definen una serie de principios o requisitos fundamentales que deben cumplir los distintos modelos de contexto, como el nivel de formalidad, la capacidad para tratar datos incompletos o ambiguos, la aplicabilidad dentro de las tecnologías existentes, o la capacidad para trabajar de forma distribuida. De entre todos los modelos analizados, los más favorecidos son los **modelos de contexto basados en ontologías**, destacando por su capacidad para trabajar de forma distribuida y su alto nivel de formalidad.

Preuveneers et al. (2004) presentan un modelo de contexto basado en ontologías adaptable y extensible dentro del ámbito de la Inteligencia Ambiental, de forma que partiendo de una ontología básica y genérica, ésta pueda extenderse para utilizarse en cualquier tipo de aplicación y dominio. Este modelo consta de 4 entidades básicas:

- **Usuario.** Las aplicaciones deben adaptarse al usuario y no al revés. Algunas propiedades del usuario son su *perfil*, sus *preferencias*, *estado de ánimo* y *actividad/tarea actual*.
- **Entorno.** El lugar donde interactúa el usuario. Contempla *tiempo* y *localización*, así como condiciones del entorno como *temperatura* y *luz*.
- **Plataforma.** Descripción del hardware y software del dispositivo del usuario, e.g., procesador, memoria, ancho de banda, y sistema operativo.
- **Servicios.** Entidades computacionales que ofrecen funcionalidades específicas a al usuario dentro del entorno.

Aunque este trabajo no se orienta dentro del ámbito de la Inteligencia Ambiental, algunos de estos contextos genéricos y algunas de sus propiedades como son las preferencias del usuario, su estado de ánimo, y la localización y tiempo actuales son relevantes a la hora de proporcionar recomendaciones contextualizadas básicas.

El modelo de contexto propuesto por Wang et al. (2004) está basado en ontologías, utilizando el lenguaje OWL como forma de representación semántica que permitir realizar inferencias y compartir conocimiento sobre el contexto, así como otros aspectos como la clasificación, dependencia y calidad del contexto. Este modelo consta de una ontología de nivel superior general, y de una serie de ontologías a un nivel inferior que aplican a distintos subdominios. El modelo incluye aspectos contextuales de primer nivel como *persona*, *localización*, *entidad computacional* y *actividad*, describiendo sus propiedades y las relaciones entre ellos, ya que esos aspectos podrían reutilizarse para describir prácticamente cualquier tipo de situación contextual. Por otro lado, presenta aspectos contextuales de nivel inferior para cubrir aplicaciones y servicios contextuales encuadrados en algún sub-dominio particular con vocabularios y conceptos de contexto específicos.

Wang et al. (2004) tratan el concepto de ontología como un mecanismo de compartición de conocimiento entre dominios, que es concebido como un conjunto de entidades, relaciones, funciones, axiomas e instancias. Los autores exponen una serie de razones por las que optar por ontologías para modelar contexto:

- **Compartición de conocimiento:** las ontologías fijan una serie de conceptos comunes para que las entidades y agentes sean capaces ‘entender’ esos conceptos de contexto mientras interactúan entre ellos.
- **Inferencia lógica:** a partir de las ontologías se pueden emplear una serie de mecanismos de inferencia lógica para obtener e.g., conceptos contextuales de alto nivel a partir de conceptos en contextos de bajo nivel. Además, permiten comprobar, localizar y solucionar inconsistencias producidas a partir de los datos obtenidos por sensores.
- **Reutilización de conocimiento:** las ontologías pueden reutilizarse en distintos dominios e incluso pueden integrarse para componer ontologías mayores.

Una extensión de este modelo, fue utilizada por Gu et al. (2005), donde lo integraron dentro de una arquitectura middleware para crear distintos servicios conscientes del contexto; y posteriormente por Ejigu et al. (2007), donde proponen un modelo basado en ontologías, también basado en OWL a la hora de modelar el contexto. Este modelo considera una serie de entidades genéricas dentro de la ontología que parten de un concepto general de ‘contexto’, y que a su vez pueden extenderse con nuevas entidades que esta vez dependan del dominio de la aplicación. Estas entidades de contexto son: *persona, dispositivo, comunicaciones, entorno físico, actividad, localización y servicio*.

Kim et al. (2012) proponen un modelo de contexto basado en ontologías aplicando lo que ellos llaman la técnica de las ‘5 Ws y 1H’, o 5W1H, para la definición y gestión de la información contextual, orientado a aplicaciones de inteligencia ambiental. Emplear un modelo es necesario para procesar la información contextual recogida desde los distintos dominios y situaciones, así como para distribuir esa información contextual a las aplicaciones que dependan de ella. Mediante la aplicación del modelo 5W1H, se pueden mapear los distintos *elementos* que una aplicación basada en contexto debe tener en cuenta, como serían el ‘rol’, el ‘objetivo’, la ‘acción’, el ‘estado’, la ‘localización’ y el ‘tiempo’, los cuales se corresponden con las preguntas ‘Who’, ‘Why’, ‘How’, ‘What’, ‘Where’ y ‘When’ respectivamente:

- **Why (objetivo):** describe cual es el objetivo último del procesado de la información contextual. En el caso de este trabajo, sería el obtener recomendaciones de ítems contextualizadas.
- **Who (rol):** describe el rol necesario para procesar la información contextual, en este caso sería el propio sistema de recomendación.

- **How (acción):** describe la acción necesaria para procesar esa información contextual, para así alcanzar el objetivo enunciado anteriormente. En este caso representaría el procesamiento de la información contextual extraída de las reviews para así realizar las recomendaciones.
- **What (estado):** representa la información contextual en sí misma. En este caso, representaría los distintos aspectos y valores de contexto extraídos de las reviews.
- **Where (localización):** representa la información de localización o posicionamiento como un contexto en sí mismo.
- **When (tiempo):** representa el tiempo, y más concretamente el intervalo temporal en que la información contextual es válida.

En el trabajo de Bettini et al. (2010) se presentan una serie de **requisitos** que deben tenerse en cuenta a la hora de realizar el modelado de contexto. Estos requisitos se tratan desde un punto de vista genérico en lo que a información contextual se refiere, es decir, sin asociarla a un dominio concreto. Estos requisitos se enumeran a continuación:

- **Heterogeneidad y movilidad:** los modelos de contexto tienen que tener en cuenta la gran variedad de información contextual que puede existir, y que ésta puede proceder de distintas fuentes de información, así como a su vez también puede venir derivada de otra información contextual. Por otro lado, muchas aplicaciones basadas en contexto se ejecutan sobre dispositivos móviles, lo que se añade al problema de la heterogeneidad; por tanto, los modelos contextuales deben estar adaptados para que el entorno varíe.
- **Relaciones y dependencias:** los modelos de contexto deben poder permitir capturar las relaciones entre la información contextual, así como las dependencias que puedan existir entre las distintas entidades.
- **Temporalidad:** las aplicaciones contextuales pueden necesitar acceder a estados pasados, por tanto, los modelos de contexto deben poder capturar y gestionar los valores históricos de la información contextual.
- **Imperfección:** la información contextual puede tener una calidad variable, incluso ser incorrecta. Por tanto, el modelo de contexto debe incluir un modelado de la calidad de dicha información.
- **Razonamiento:** los modelos de contexto deben permitir aplicar técnicas de razonamiento sobre ellos, para así poder encontrar nuevos contextos, o contextos de más alto nivel.
- **Formalidad:** el modelo de contexto debe tener cierto nivel de formalidad, disponiendo de estructuras claras y fáciles de manipular por las aplicaciones, así como fáciles de trasladar información del mundo real al modelo.

- **Eficiencia:** el modelo de contexto debe permitir un acceso eficiente a la información contextual que contempla; algo que puede ser complejo si se disponen de un número grande de objetos. Habitualmente, se entiende la existencia de ciertos contextos primarios (como pueden ser localización, el tipo de objeto, el tiempo o la actividad), y contextos secundarios que derivan de contextos primarios.

También describen y evalúan una serie de modelos de contexto genéricos, es decir, que puedan servir para cualquier tipo de aplicación, independientemente del dominio de esta. En concreto, se evalúan tres tipos: modelos basados en objetos y rol; modelos espaciales; y modelos basados en ontologías. Analizando estos tres modelos teniendo en cuenta los requisitos enumerados anteriormente, se puede ver como ninguno de los tres satisface todos los requisitos, sino que cada modelo tiene sus puntos fuertes y débiles, por ejemplo, los modelos basados en ontologías destacan en cuanto a la heterogeneidad y el uso de relaciones y dependencias, mientras que por el contrario tienen carencias a nivel de temporalidad o de movilidad. Por esta razón, Bettini et al. (2010) hablan de la necesidad de contar con soluciones híbridas en cuanto a modelado de contexto, que combinen varias de estas técnicas para así poder suplir las carencias de una determinada técnica con los puntos fuertes de otra.

Finalmente, Krummenacher y Strang (2007) describen una serie de criterios y características en base a las cuales deben construirse los modelos de contexto basados en ontologías. Estos criterios son importantes, ya que, a la hora de evaluar distintos modelos basados en ontologías, se debe contar con un marco común de referencia de forma que se pueda medir cada modelo bajo los mismos criterios. En su trabajo, distinguen dos tipos de criterios: por un lado, aplicados al modelado de contexto, como pueden ser la calidad de los datos, su aplicación dentro de la infraestructura actual, su trazabilidad, su capacidad para comparar datos, y su nivel de inferencia; y por otro lado, criterios aplicados directamente para evaluar la ontología, como pueden ser su flexibilidad, su capacidad para ser extendida, si es genérica o específica, su nivel de consistencia, su escalabilidad, y su nivel de formalidad.

## 3.2 Reconocimiento de información contextual en texto

En la mayoría de las publicaciones sobre sistemas de recomendación conscientes del contexto, el contexto se define como un conjunto limitado de variables discretas que pueden ser observadas y cuya estructura (atributos, valores, etc.) se conoce a priori y no cambia con el tiempo. Esto, sin embargo, no se corresponde con muchos casos reales, en los que la información contextual no es conocida a priori, sino que tiene que ser inferida o incluso modelada a partir de los datos.

Una parte de los sistemas conscientes del contexto, se encuadran dentro del ámbito de la Computación Ubicua (Hoseini-Tabatabaei et al., 2013), por lo que el contexto es extraído empleando sensores físicos, tales como sensores de movimiento (acelerómetros, giroscopios), sensores de posicionamiento (GPS, WIFI) o sensores ambientales (cámara, micrófono) capaces de capturar los distintos elementos contextuales, para después llevar a cabo un procesamiento de esta información y poder extraer las características contextuales tales como el tipo de actividad que se está desarrollando, identificar interacciones sociales o identificar el entorno para después emplearlo en beneficio del usuario a través de un sistema o aplicación contextual. No obstante, este trabajo se centra en situaciones donde la información contextual tiene que extraerse de reviews textuales sobre ítems proporcionadas por usuarios.

Bauman y Tuzhilin (2014) proponen un método para extraer esa información contextual de reviews. Como indican en su trabajo, no todas las reviews contienen información contextual, o si la tienen, no es de calidad. Por ello, el primer paso que siguen es separar las reviews en dos subconjuntos: reviews genéricas que expresan impresiones generales y contienen información contextual muy limitada, y reviews específicas que contienen información contextual concreta, e.g., el momento exacto al que una review se refiere sobre cuándo el usuario usó o consumió un ítem. Esto lo hacen etiquetando previamente cada una de las reviews con una serie de mediciones que ponderan ciertos aspectos, como el número de frases, el número de palabras, y el número de verbos en pasado. A continuación, basándose en ciertos supuestos como que las reviews genéricas tienden a ser más cortas que las reviews específicas y suelen expresarse en pasado, usan esas mediciones para caracterizar las reviews. En base a esas mediciones, los autores usan técnicas de clustering (e.g., *K-means*) para separar las reviews en específicas y genéricas. Tras separarlas, los autores identifican qué términos (en concreto nombres) aparecen de forma más frecuente, y empleando WordNet (Miller, 1995) identifican los ‘*synsets*’ o significados a los que pertenecen esas palabras. Entre esos *synsets* identifican las distintas categorías contextuales que se quieren explotar.

Para evaluar sus resultados, Bauman y Tuzhilin llevan a cabo una anotación manual de distintas reviews, identificando qué contextos aparecen en cada una de ellas. Después, ordenando la lista de *synsets* obtenida de forma automática y repasándola manualmente, consiguen identificar algunos contextos similares a los que existían en las anotaciones manuales, como el tiempo del día, el día de la semana, la compañía del usuario, si se trata de una ocasión especial (e.g., cumpleaños), o si el usuario conocía un sitio antes de visitarlo. De esta forma, pueden analizar cómo el proceso de anotación es capaz de identificar distintos contextos presentes en las reviews, donde el problema es determinar cuántos grupos de la lista es necesario recorrer hasta descubrir las variables de contexto que se quieren identificar.

La extracción de información contextual no es una tarea sencilla, ya que por un lado hay que trabajar con datos no estructurados expresados en lenguaje natural –con toda

la diversidad, ambigüedad y ruido que conlleva-, y por otro lado hay que identificar los aspectos y valores contextuales en el texto que pueden expresarse por los usuarios de múltiples formas y tal vez de manera indirecta. Por eso, una parte importante de este trabajo consiste en identificar qué textos contienen información contextual.

Aciar (2010) presenta un método capaz de detectar en reviews de usuarios frases que contienen información contextual. Para ello, el método emplea técnicas de minería de textos y clasificación, y establece una serie de reglas de decisión construidas de forma automática que detectan las frases con información contextual. Sin embargo, se limita a identificar dichas frases, y no a extraer la información contextual existente en ellas. Para llevar a cabo el entrenamiento del sistema, Aciar anotó manualmente las frases de 100 reviews extraídas de TripAdvisor<sup>8</sup>, indicando cuales contenían contexto, para después procesarlas y generar un conjunto de reglas de decisión encargado de identificar nuevas frases de contexto, e.g., aquellas que poseen la palabra ‘cumpleaños’. Con el nuevo repositorio de frases, evaluó su método sobre las frases contenidas en otras 50 reviews. Estas técnicas de minería de texto y la posterior extracción de subconjuntos de reglas de decisión, también descritas y aplicadas por Aciar et al. (2007), si bien no permiten extraer la información contextual concreta, sí permiten identificar qué frases contienen información contextual, por lo que el método se podría emplear como un pre-filtrado de frases de reviews, que permitiría posteriormente aplicar técnicas de extracción de contexto únicamente sobre las frases identificadas como portadoras de contexto.

En el trabajo llevado a cabo por Li et al. (2010) se emplean técnicas de procesamiento de lenguaje natural para extraer información contextual de reviews de usuario sobre restaurantes, centrándose en cuatro tipos de contexto: *tiempo* (momento del día para acudir a comer, e.g., desayuno, almuerzo, comida, cena), *ocasión* (motivo por el que se acudió al restaurante, e.g., cumpleaños, aniversario, reunión de empresa), *localización* (lugar o zona en la que se encuentra el restaurante) y *compañía* (personas que acompañaron al usuario al restaurante). Para cada uno de estos contextos, los autores plantean una manera diferente de extracción de información contextual, aunque todas ellas están basadas en técnicas de minería de textos. Por ejemplo, para extraer la información contextual referente al contexto del tiempo, se emplea un *lexicón* con los distintos tipos de momentos temporales para ir a un restaurante, y estos se buscan dentro del texto de la reviews. Algo similar se realiza para el contexto de compañía; se buscan patrones textuales (e.g., ‘con mi (mis) + sustantivo’) que indiquen la presencia de otras personas como familiares y amigos. En el mismo trabajo, los autores finalmente comprueban que incluir información contextual dentro de un sistema de recomendación ayuda a obtener mejores recomendaciones, ya que dicha información

---

<sup>8</sup> TripAdvisor, portal web de viajes, <http://www.tripadvisor.com>

efectivamente influye a la hora de tomar decisiones en el dominio de restaurantes. Además, muestran que la extracción de información contextual es una componente importante del proceso de generación de recomendaciones, ya que el hecho de extraer información contextual de mayor calidad hace que las recomendaciones sean más efectivas.

Lahlou et al. (2013) también aplican diversas técnicas de minería de textos para intentar extraer el contexto de reviews de usuarios. Antes de describir las técnicas, enumeran una serie de retos que existen a la hora de encontrar información contextual en reviews de ítems, como pueden ser los errores ortográficos cometidos, la pequeña longitud de los textos, y en ocasiones una gran diversidad léxica en cuanto a la variedad de términos empleados para referirse a un contexto dado. También se da la circunstancia de que habitualmente cuando un usuario escribe una review, no lo hace con el objetivo de describir información contextual, sino simplemente de valorar su experiencia, por eso, a la hora de querer inferir dicha información contextual, es posible que uno se encuentre con numeroso “ruido”, con nula o escasa información útil sobre contexto. Para llevar a cabo la tarea de extraer dicha información, Lahlou et al. tratan las reviews como si fuesen documentos de texto y aplican técnicas de “tokenización” (i.e., transformación el texto en una secuencia de *tokens* o palabras separadas por espacios en blanco) en unigramas, bigramas o trigramas, dependiendo de si se separan grupos de una, dos o tres palabras, respectivamente; técnicas de *stemming*, que consisten en reducir una palabra a su raíz morfológica (en inglés *stem*) con el fin de unificar y reducir el número de términos del modelo, agrupando palabras similares bajo su misma raíz; y una técnica que consiste en establecer ciertos límites de frecuencia de aparición de palabras, para eliminar aquellas palabras que sobrepasen esos límites. Esta técnica tiene sentido, ya que, si una palabra aparece poco en las reviews, se puede decir que no es informativa, y por lo tanto puede ser descartada. Similarmente, si una palabra aparece en un alto número de documentos, también puede eliminarse, ya que, al ser tan común, es poco discriminativa. Aplicadas estas técnicas, cada review se modela en una representación vectorial de los distintos términos, y usando un modelo TF-IDF se puede determinar el peso de cada término dentro de cada review. Después, mediante algún algoritmo de clasificación como Naïve Bayes (NB) o Maquinas de Vectores Soporte (*Support Vector Machine* o *SVM*) se puede determinar las reviews que tienen o carecen de información contextual. En las diversas pruebas llevadas a cabo, Lahlou et al. concluyen que para los conjuntos de datos empleados, el modelo de clasificación por Naïve Bayes da mejores resultados que SVM. Además, la mejor combinación de técnicas de tratamiento de textos empleadas fue la de eliminar aquellos términos que sólo aparezcan una vez; combinar unigramas, bigramas y trigramas para confeccionar los posibles términos de los que se conforma el modelo; después aplicar *stemming* y por último obtener el peso de cada término en cada review utilizando TF-IDF.

Finalmente, otra aproximación interesante para realizar la tarea de anotación de reviews con información contextual es la llevada a cabo por Hariri et al. (2011). En su trabajo aplican técnicas de aprendizaje automático supervisado para clasificación multi-clase, empleando *Labeled LDA (Latent Dirichlet Allocation)*, donde una review puede ser asignada a las distintas clases con cierta probabilidad. Esta forma de inferir información contextual se acompaña de un algoritmo propio de recomendación de tipo ‘modelado contextual’, que emplea vectores con las probabilidades de pertenencia de las reviews a cada una de las clases de contexto. En sus pruebas, observan como desde el punto de vista del *hit-ratio*, su modelo ofrece mejores resultados que los sistemas de recomendación que no tienen en cuenta el contexto, como el kNN basado en ítem, y muestra de nuevo que la extracción, ya sea de manera explícita o implícita, y explotación de contexto son beneficiosas a la hora de generar mejores recomendaciones.

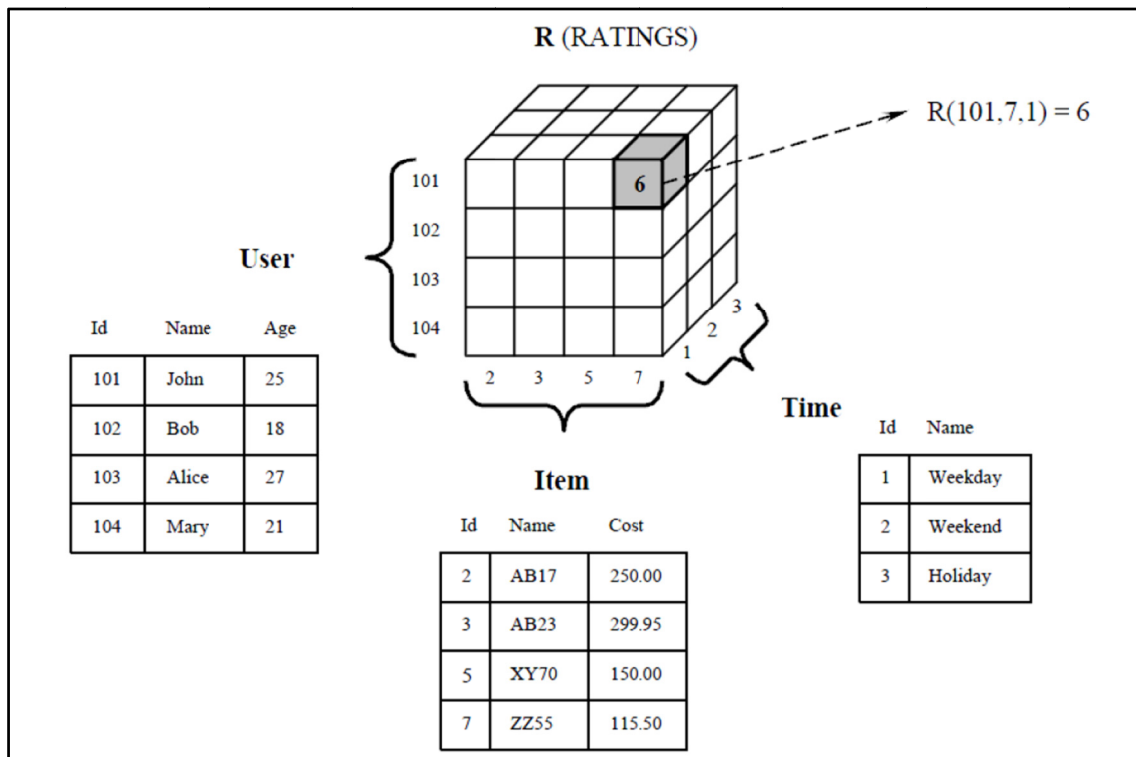
### 3.3 Recomendación contextualizada

Los sistemas de recomendación conscientes del contexto (CARS, *Context-Aware Recommender Systems*) adaptan recomendaciones personalizadas a las situaciones específicas donde los ítems recomendados serán consumidos por los usuarios, a través de la explotación de información contextual, e.g., momento temporal, localización, compañía y estado de ánimo actuales del usuario. Para incorporar esa información en el proceso de generación de recomendaciones, Adomavicius et al. (2005) proponen un marco de recomendación genérico multidimensional, que establece un espacio con múltiples dimensiones asociadas a los usuarios, ítems y aspectos contextuales. Formalmente, sean  $D_1, D_2, \dots, D_n$  tales dimensiones. Con ellas se define el espacio de recomendaciones como el producto cartesiano de todas ellas. De esta forma, la función de utilidad quedaría definida de la siguiente manera:

$$f(u, i, C) = R : D_1 \times D_2 \times \dots \times D_n \rightarrow \text{Ratings}$$

donde  $C$  representa las variables contextuales usadas. Por ejemplo, si se considera un modelo de recomendación en el espacio tridimensional  $Usuarios \times Items \times Tiempo$ , la función  $R$  significaría el rating que el usuario  $u \in Usuarios$  daría al ítem  $i \in Items$  en el momento del tiempo  $t \in Tiempo$ ,  $R(u, i, t)$ . Esto podría representarse de forma gráfica como un modelo cúbico tridimensional, como se muestra en la Figura 3.1, donde  $R(101, 7, 1) = 6$  representa que para el usuario con ID 101 y el ítem con ID 7, se asignó una puntuación de 6 durante los días laborales de la semana.





**Figura 3.1** Representación multidimensional propuesta por Adomavicius et al. (2005) para realizar recomendaciones contextualizadas

En esta representación multidimensional, para llevar a cabo la recomendación, Adomavicius et al. (2005) proponen, de manera genérica, llevar a cabo una reducción de los datos a dos dimensiones, usuarios e ítems, para así poder emplear las técnicas tradicionales de recomendación. Esta técnica de reducción de dimensionalidad consiste en pre-procesar los datos y emplear sólo aquellos que coinciden con las condiciones contextuales actuales. Por ejemplo, si en el modelo de la figura anterior se quisiera recomendar ítems para el fin de semana, se seleccionarían sólo los ratings que se corresponden el valor *'weekend'* de la dimensión temporal, desechando el resto (i.e., los pertenecientes a *'weekday'* y *'holiday'*). Una vez hecho esto, como la dimensión contextual tendría el mismo valor para todos los ratings (en este caso, fin de semana), se tendría un conjunto de datos de dos dimensiones (usuarios e ítems), sobre los que se podrían aplicar técnicas tradicionales de recomendación, e.g., basadas en contenido o de filtrado colaborativo.

En ese mismo artículo, Adomavicius et al. muestran empíricamente que en situaciones donde el contexto es importante, los sistemas de recomendación conscientes del contexto dan mejores recomendaciones que sistemas tradicionales que usan todos los ratings disponibles, sin tener en cuenta sus contextos.

De manera más específica, Palmisano et al. (2008) analizan cómo de importante es la información contextual a la hora de construir modelos predictivos (en el caso particular de clientes de una tienda), además de estudiar si es posible inferir esa información

contextual a partir de los datos de los que disponían. Los autores muestran que el contexto es importante a la hora de modelar a los usuarios de manera individual (i.e., empleando un modelo distinto para cada uno de los usuarios), pero que esa importancia pierde fuerza a medida que se van agregando los distintos usuarios en segmentos dependientes del contexto. También presentan un método Bayesiano para inferir información contextual a partir de datos no estructurados, y emplean la información contextual obtenida para mejorar un modelo de recomendación no contextualizado.

El modelo basado en Factorización de Tensores (*Tensor Factorization*) es uno de los que mejores resultados ha alcanzado en los estudios publicados sobre CARS (Karatzoglou et al., 2010). Consiste en extender un modelo original de dos dimensiones (*Usuarios x Items*) a n-dimensiones aplicando el uso de tensores. El modelo introduce un gran número de parámetros que deben ser establecidos durante la fase de entrenamiento, y además, tal y como indican los autores, el número de esos parámetros aumenta a medida que hay más factores contextuales. Así, para que el método produzca buenos resultados, es necesario que cuente con una cantidad de datos de entrada, con los que calcular adecuadamente dichos parámetros. Si por el contrario no se dispone de una gran cantidad de datos, los autores comentan que es mejor escoger modelos más simples, evitando el sobreajuste (*overfitting*). Tal y como indican Baltrunas et al. (2011), es necesario llegar a un equilibrio entre la complejidad del modelo y el número de datos de entrenamiento disponibles.

Siguiendo esta idea, Baltrunas et al. (2011) proponen tres modelos mucho más simples que *Tensor Factorization*, consistentes en extensiones del popular modelo Factorización de Matrices (*Matrix Factorization*, *MF*) con información contextual de los ratings: *Context-aware Matrix Factorization*, CAMF. Cada uno de estos tres modelos realiza distintos supuestos a la hora de tratar la información contextual junto a los *ratings*:

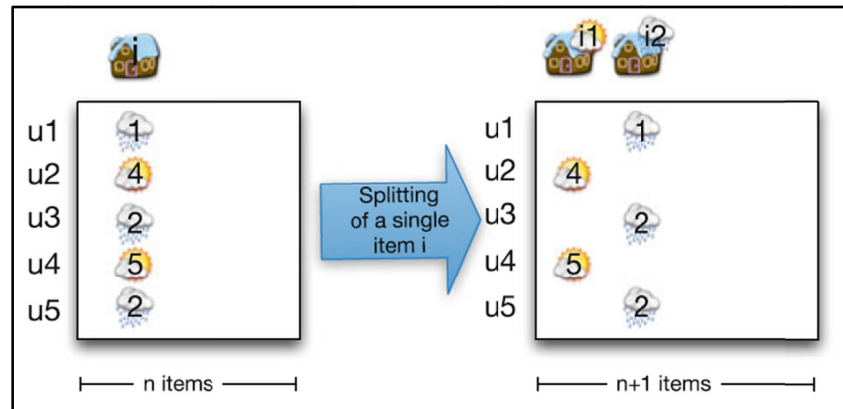
- **CAMF-C:** es el modelo más simple de los propuestos (i.e., con el menor número de parámetros), y asume que los factores contextuales tienen una influencia uniforme sobre los *ratings*, sin importar los ítems en sí mismos. Esto significa que, si por ejemplo se tiene el factor contextual del tiempo meteorológico, y toma valor de ‘soleado’, se produce un impacto similar en el valor de los ratings de todos los ítems (e.g., todos los lugares de interés aumentan su rating por estar soleado). Este modelo introduce un parámetro por cada condición contextual.
- **CAMF-CI:** es el modelo más complejo de los propuestos (i.e., con el mayor número de parámetros), y asume que cada factor contextual influye de manera diferente a cada ítem de manera individual. Por tanto, se introducen tantos parámetros como pares ‘condición contextual’-‘ítem’ haya. Esto se traduce en que por ejemplo si el valor contextual ‘tiempo meteorológico’ toma el valor de

‘soleado’, esto puede influir en el *rating* de un ítem con ese contexto, pero no de otros.

- **CAMF-CC**: es un modelo intermedio en cuanto a complejidad. Introduce un parámetro por cada condición contextual y categoría de ítem existentes. De esta forma, se asume que los ítems pueden agruparse en categorías (e.g., los lugares de interés pueden clasificarse en museos, plazas...), en lugar de tratarlos de manera individual, y las condiciones contextuales tienen el mismo impacto sobre todos los ítems de una categoría.

En los experimentos presentados en (Baltrunas et al., 2011) sobre dos datasets, uno real y el otro generado de forma semi-automática, se compara el método de *Tensor Factorization* con el método CAMF, en particular con el modelo CAMF-CI. Este último obtuvo mejores resultados en cuanto a menor MAE. Posteriormente, en el mismo trabajo, los autores comparan los tres modelos entre sí, además de la técnica tradicional de Factorización de Matrices (MF) sin contexto. En este caso, los tres modelos propuestos obtienen mejores resultados en cuanto a menor MAE con respecto a MF. Al comparar los modelos, los mejores resultados son obtenidos por el modelo intermedio CAMF-CC en ambos datasets.

Otra técnica muy conocida en la literatura de CARS es la denominada *Item Splitting* (Baltrunas & Ricci, 2009 y 2014). Esta técnica es parecida al modelo de reducción explicado anteriormente (Adomavicius et al., 2005) en cuanto a que también pertenece a la categoría de pre-filtrado de contexto, pues modifica los datos iniciales en base a información contextual. Sin embargo, sus diferencias (tanto en implementación como en resultados) son notables. La idea general de *Item Splitting* es que los ratings de determinados ítems son divididos en dos subconjuntos de acuerdo con el valor de un determinado factor contextual. Por ejemplo, los ratings de cierto lugar de vacaciones pueden separarse entre aquellos ratings que fueron proporcionados en invierno, y aquellos dados en verano. Así, ese ítem se representa como dos pseudo-ítems, que tienen asociados los valores contextuales ‘invierno’ y ‘verano’. A cada uno de estos pseudo-ítems se les asigna entonces el subconjunto de ratings correspondiente, y los dos subconjuntos de ratings se integran con el resto de ítems del sistema. En la siguiente figura puede verse contemplado este proceso:



**Figura 3.2** Un ítem se separa en dos pseudo-ítems que tienen asociados los distintos valores contextuales y el subconjunto de ratings correspondiente y se integran de nuevo en el sistema (Baltrunas & Ricci, 2014)

Esta división se realiza sólo si existe una evidencia estadística de que los subconjuntos de ratings generados son distintos, es decir, si el valor contextual por el que se han separado los ratings del ítem realmente influye a la hora de generar recomendaciones. Más específicamente, para determinar si cada factor contextual debe emplearse para dividir ítems, se aplican ‘criterios de impureza’ (*impurity criteria*) que eligen aquellos factores contextuales que maximicen cierta métrica estadística. Para cada factor contextual se generan las correspondientes divisiones de ratings, y para ellas se mide su valor en cierto criterio de impureza. Si ese valor supera cierto umbral, se determina que la separación de ratings producida por el factor contextual es relevante, y se mantiene la división; en caso contrario, la división se descarta. De esta forma, se asegura el mantenimiento de una significancia estadística en la diferencia de subconjuntos de ratings generados. Algunos de los criterios de impureza empleados por Baltrunas et al. se basan por ejemplo en calcular cómo de significativa es la diferencia entre las medias de los subconjuntos de ratings, o cómo de significativa es la proporción de ratings altos y bajos en ambos subconjuntos, o midiendo el valor de la divergencia de la información (*Information Gain*), también conocida como divergencia de Kullback-Leibler (Quinlan, 2014).

En sus experimentos, Baltrunas et al. observan que *Item Splitting* resulta beneficioso si los factores contextuales usados generan subconjuntos de ratings más homogéneos que el conjunto de ratings original. Otro resultado mostrado por los autores es que al aplicar la técnica de Item Splitting, la lista de recomendaciones generada es distinta que la lista obtenida empleando todos los ítems originales, y que a mayor impacto de los factores contextuales en los ítems, mayor es la variación entre ambas listas. No obstante, el estudio realizado no entra en valorar cómo de eficaces son esas nuevas listas en función de la menor o mayor dependencia del contexto, dejándolo como trabajo futuro.

La técnica de *Item Splitting* puede modificarse para en vez de dividir los ítems en pseudo-ítems, dividir a los usuarios, dando lugar a la técnica denominada como *User Splitting*. También se puede aplicar una combinación de ambas técnicas, es decir, dividir tanto los ítems como los usuarios. Esta última técnica se conoce como *UI Splitting*. Zheng et al. (2014) llevaron a cabo una comparación empírica de estas tres técnicas de *splitting*, comparándolas además con otros algoritmos de filtrado colaborativo conscientes del contexto. En su artículo observan como norma general que la técnica combinada de *UI Splitting* ofrece mejores resultados (en términos de RMSE), siempre y cuando se emplee el criterio de impureza adecuado. También comentan que según la dependencia que exista entre el contexto y los usuarios/ítems, puede ser más preferible emplear la técnica de *Item Splitting* o la de *User Splitting*. Por ejemplo, si se usa un *dataset* con reviews de restaurantes, el contexto ‘compañía’ está más ligado a los usuarios que a los ítems, por lo que aplicar *User Splitting* puede ser más beneficioso que aplicar *Item Splitting* si se considera ese factor contextual. Los autores no concluyen con un criterio de impureza óptimo, y comentan que ése dependerá de los datos y que deberá ser establecido durante la fase de entrenamiento.

Según lo expuesto, las técnicas de *splitting* pueden considerarse como buenas alternativas a otros CARS existentes.

Recientemente, Codina et al. (2016) han presentado una nueva técnica de pre-filtrado contextual. Esta técnica, denominada DSPF (*Distributional-Semantics Pre-Filtering*), representa una novedosa y prometedora forma de construir modelos de predicción de ratings contextualizados. Su idea general es que para un contexto dado, se construye un modelo de predicción utilizando aquellos ratings que tengan contextos semánticamente similares. Después, ese modelo ‘local’ construido a partir de los contextos semánticamente similares se emplea para hacer predicción sobre el contexto dado originalmente, y para realizar recomendaciones sobre esa situación contextual, aplicando en este caso Factorización de Matrices (DSPF-MF). El concepto de similitud entre contextos se establece en función de la influencia que tienen sobre los ratings. Codina et al. hablan de modelos ‘locales’, ya que no están empleando todos los datos del dataset, sino sólo una parte obtenida a partir de los contextos similares al contexto dado; de esta forma, se entiende que, aunque se empleen menos datos, estos resultan de mayor calidad, pues son más relevantes para la situación contextual actual. En la evaluación llevada a cabo sobre distintos datasets, y comparando DSPF-MF con otros algoritmos considerados como el estado del arte, como son CAMF (*Context-Aware Matrix Factorization*) y *UI Splitting*, los autores observan como DSPF-MF da mejores resultados en términos de MAE y nDCG, métrica que tiene en cuenta no sólo que se recomiende un mayor número de ítems relevantes, sino que estos aparezcan en las primeras posiciones de las listas de recomendación.



## Capítulo 4

# Solución desarrollada

En este capítulo se presenta la solución desarrollada para llevar a cabo la extracción de información contextual de valoraciones textuales (reviews) escritas por usuarios sobre ítems de diferentes dominios, para después emplearla por métodos de recomendación contextualizada. En primer lugar, se detalla la construcción de una taxonomía genérica de contexto, generada a partir de entidades extraídas de DBpedia; posteriormente se describe el proceso de anotación contextual de las reviews empleando dicha taxonomía; y por último, se explican los algoritmos de recomendación contextualizada evaluados con las anotaciones de contexto generadas.

### 4.1 Taxonomía de contexto a partir de DBpedia

En esta sección se detalla la taxonomía de contexto genérica construida a partir de entidades semánticas extraídas de DBpedia. En primer lugar, se describe la taxonomía que se ha generado, indicando las cuatro dimensiones de contexto principales que abarca. Después, se detalla el proceso de construcción automática de la taxonomía a partir de entidades extraídas de DBpedia. Por último, se presenta una herramienta y proceso empleados para llevar a cabo una edición semi-automática de la taxonomía.

#### 4.1.1 Categorías principales de la taxonomía

Una taxonomía puede definirse como la clasificación/categorización de un conjunto de objetos, en este caso tipos de contexto, de forma jerárquica y empleando distintos niveles de profundidad, de forma que a medida que se avanza en su jerarquía a niveles más profundos, los objetos pasan de conceptos más genéricos a conceptos más específicos.

En este caso, la taxonomía de contexto representará las distintas categorías de contexto de una forma jerarquizada, partiendo de un nodo raíz que podríamos considerar sencillamente como ‘contexto’ hasta niveles más profundos que identifican categorías contextuales más particulares, como podría ser una categoría que represente el momento del día entendido como contexto.

Para llevar a cabo la construcción de la taxonomía, en primer lugar, es necesario determinar las dimensiones contextuales que va a abarcar. Estas dimensiones pueden entenderse como contextos de primer nivel, y representan cada una de las ramas principales de la taxonomía. Para determinar esas dimensiones se han consultado algunas de las representaciones de contexto existentes en la literatura y revisadas en la sección anterior. En la siguiente tabla se muestran cuáles son las principales categorías contextuales empleadas en esos trabajos.

Referencia	Categorías de contexto
<i>Abowd et al. (1999)</i>	<b>Contextos principales:</b> identidad (quien es el usuario), ubicación (donde está el usuario), actividad (que está haciendo el usuario), tiempo (cuando lo está haciendo)  <b>Contextos secundarios:</b> extensiones de estos contextos primarios
<i>Preuveneers et al. (2004)</i>	<b>Cuatro entidades básicas:</b> usuario (perfil, preferencias, estado de ánimo), entorno (tiempo, localización), plataforma (hardware y software), servicios
<i>Wang et al. (2004)</i>	<b>Ontología a nivel general con cuatro contextos:</b> persona, localización, entidad computacional, actividad.
<i>Gu et al. (2005)</i>	Ontologías a nivel inferior: extienden las anteriores, distintos subdominios.
<i>Ejigu et al. (2007)</i>	<b>Ontología a nivel general con siete contextos:</b> persona, dispositivo, comunicaciones, entorno físico, actividad, localización y servicio.  Ontologías a nivel inferior: extienden las anteriores, distintos subdominios.
<i>Li et al. (2010)</i>	<b>Cuatro tipos de contexto</b> extraídas de reviews de restaurantes: tiempo (momento del día para acudir), ocasión (motivo por el que se acude al restaurante), localización (lugar o zona), compañía (personas que acompañan al usuario)
<i>Kim et al. (2012)</i>	<b>Modelo 5W1H:</b> compuesto de objetivo ( <i>Why</i> ), rol ( <i>Who</i> ), estado ( <i>What</i> ), localización ( <i>Where</i> ), tiempo ( <i>When</i> ) y acción ( <i>How</i> )

**Tabla 4.1** Categorías de contexto empleadas por algunos modelos ontológicos.

Como se puede observar, algunas de estas categorías aparecen representadas en varios de los modelos ontológicos descritos en la tabla, por lo que pueden ser susceptibles de representar dimensiones de contexto en la taxonomía. Por otro lado, para el caso de este trabajo, se ha tenido en cuenta la idea de querer construir una taxonomía de



contexto genérica, es decir, que sea independiente del dominio donde se aplique, por lo que las dimensiones de contexto seleccionadas deben ser lo suficientemente genéricas.

En particular, se ha decidido emplear cuatro dimensiones contextuales principales que representan las cuatro ramas principales de la taxonomía. Las cuatro dimensiones contextuales escogidas son el contexto de **tiempo** (*Time context*) que representa el tiempo o momento en el que se lleva a cabo la acción; el contexto de **localización** (*Location context*) que representa el lugar donde se lleva a cabo la acción; el contexto de **entorno** (*Environmental context*) que representa lo que rodea a la acción; y el contexto **social** (*Social context*) que representa lo que rodea al usuario, tanto sus emociones como la compañía.

A partir de estas cuatro dimensiones, se han seleccionado categorías en DBpedia que representan esos conceptos, para servir como categorías raíces a la hora de extraer las entidades semánticas relacionadas, y, por tanto, de extraer las distintas categorías contextuales que van a formar parte de la taxonomía.

#### 4.1.2 Construcción automática de la taxonomía

Una vez seleccionadas las cuatro dimensiones principales que formaran parte de la taxonomía de contexto, es necesario incluir las distintas categorías que formarán parte de cada una de ellas. Realizar este proceso manualmente resultaría costoso, por lo que se propone utilizar repositorios semánticos existentes actualmente en la Web, gracias al proyecto Linked Open Data. En concreto, se va a utilizar el repositorio de entidades semánticas *DBpedia*, que como ya se ha explicado anteriormente, representa una ontología estructurada creada a partir de la información que se encuentra en Wikipedia.

Más específicamente, para llevar a cabo la tarea de la construcción de la taxonomía, se lleva a cabo un recorrido automático del grafo semántico de DBpedia para ir obteniendo las distintas categorías contextuales vinculadas a las distintas dimensiones de contexto.

Para realizar este recorrido a través de DBpedia es necesario definir primero algunas categorías raíz por las que empezar dicho recorrido trasversal a través del grafo. Estas categorías raíz pueden verse como categorías contextuales que se encuentran justo en el siguiente nivel de las dimensiones de contexto antes citadas, ya que dependen directamente de ellas.

En la siguiente tabla se muestran las distintas categorías raíz escogidas para cada una de las cuatro dimensiones contextuales consideradas en la taxonomía.

Dimensión contextual	Categoría raíz DBpedia
<i>Time_context</i>	dbc <sup>9</sup> :Periodic_phenomena
	dbc:Working_time
<i>Location_context</i>	dbc:Places
	dbc:Earth
	dbc:Types_of_roads
	dbc:Academic_institutions
	dbc:Transport_by_mode
	dbc:Wheeled_vehicles
<i>Environmental_context</i>	dbc:Meteorological_phenomena
	dbc:Entertainment
	dbc:Domesticated_animals
<i>Social_context</i>	dbc:Interpersonal_relationships
	dbc:Emotions

**Tabla 4.2** Categorías raíz de DBpedia elegidas para las dimensiones contextuales.

Así, empezando por estas categorías, se puede ir recorriendo las distintas subcategorías en Wikipedia/DBpedia de una manera iterativa, e ir obteniendo los distintos nodos que formarán parte de la taxonomía general de contexto. Una característica importante de DBpedia, es que gran parte de sus datos están expresados mediante el vocabulario SKOS (*Simple Knowledge Organization System*), el modelo estándar de W3C para las taxonomías, por lo que a través de las distintas relaciones SKOS se puede realizar un recorrido de las categorías de DBpedia de forma jerárquica (i.e. categorías-subcategorías), para ir construyendo la taxonomía.

En este recorrido iterativo se hace uso de las distintas propiedades que enlazan categorías en DBpedia, obteniendo las distintas subcategorías (niveles más profundos de la taxonomía) y las distintas clases e instancias que forman parte de una categoría. En particular, esto se consigue gracias a la propiedad ‘*skos:broader*’ que permite identificar conceptos más genéricos y por tanto obtener diferentes subcategorías como aquellas cuya propiedad ‘*skos:broader*’ apunta a la categoría padre; y de la propiedad ‘*dct:subject*’ que permite obtener las distintas instancias y clases representadas por la categoría.

---

<sup>9</sup> ‘dbc:’ representa ‘<http://dbpedia.org/page/Category:>’, por tanto, ‘dbc:Periodic\_phenomena’ equivale a la URL ‘[http://dbpedia.org/page/Category:Periodic\\_phenomena](http://dbpedia.org/page/Category:Periodic_phenomena)’

Por ejemplo, para la categoría raíz *‘dbc:Interpersonal\_relationships’*, se han obtenido tres subcategorías que tienen ese valor en su propiedad *‘skos:broader’*, como pueden ser *‘dbc:Family’*, *‘dbc:Friendship’* y *‘dbc:Intimate\_relationships’*. De la misma manera, a partir de *‘dbc:Family’*, se han obtenido sus subcategorías como son *‘dbc:Childhood’*, *‘dbc:Housewives’*, *‘dbc:Parenting’*, *‘dbc:Sibling’* y *‘dbc:Spouses’*, y así sucesivamente hasta ir obteniendo los distintos niveles de la taxonomía. Por otro lado, por ejemplo para la subcategoría *‘dbc:Family’* a través de la propiedad *‘dct<sup>10</sup>:subject’*, se han obtenido las instancias *‘dbr<sup>11</sup>:Son’*, *‘dbr:Spouse’*, *‘dbr:Mother’*, *‘dbr:Father’*, *‘dbr:Daughter’*, etc. que se añaden como información adicional a la categoría contextual de la taxonomía. Todo este proceso de obtención de categorías e instancias es realizado de manera automática mediante consultas SPARQL sobre la base de conocimiento de DBpedia.

Después de llevar a cabo este proceso iterativo para ir recorriendo las distintas categorías e ir obteniendo las distintas subcategorías e instancias, se ha observado que además de ellas, existen otras palabras que también describen valores contextuales, pero que no están representadas ni por las categorías, ni por sus correspondientes instancias, sino que, en la mayoría de los casos, estaban representadas por sinónimos.

Por ello, para dotar de mayor profundidad léxica a la taxonomía, se ha decidido incluir sinónimos de las categorías y de las propias instancias como información adicional de las propias categorías contextuales. Para incluir estos sinónimos de categorías e instancias se ha empleado la base de datos léxica **WordNet** (Miller, 1995), a través de la cual se pueden obtener sinónimos de palabras. Estos sinónimos se obtienen automáticamente mediante el uso de la API Java JAWS.

Por otro lado, además de las propiedades de DBpedia mencionadas con anterioridad, se ha observado que hay conceptos que no están descritos a través de las instancias, pero sí están descritos en otras entidades que pueden obtenerse a través de la propiedad *‘dbo<sup>12</sup>:wikiPageRedirects’*, que representa entidades de DBpedia que redirigen (equivalen) a las distintas instancias de la categoría. Por tanto, tiene sentido que estas palabras, así como sus sinónimos, formen parte de la información adicional de las distintas categorías de la taxonomía.

Llegados a este punto, la taxonomía quedaría formada por una estructura jerárquica donde cada uno de sus nodos representa una categoría contextual que a su vez cuenta

<sup>10</sup> *‘dct:’* representa *‘http://purl.org/dc/terms/’*, por tanto, *‘dct:subject’* equivale a *‘http://purl.org/dc/terms/subject’*

<sup>11</sup> *‘dbr:’* representa *‘http://dbpedia.org/resource/’*, por tanto *‘dbr:Son’* equivale a *‘http://dbpedia.org/resource/Son’*

<sup>12</sup> *‘dbo:’* representa *‘http://dbpedia.org/ontology/’*, por tanto *‘dbo:wikiPageRedirects’* equivale a *‘http://dbpedia.org/ontology/wikiPageRedirects’*

con una serie de palabras a modo de información adicional como son los sinónimos de la propia categoría contextual, las instancias extraídas de DBpedia, los sinónimos de esas instancias, los conceptos que redirigen a las instancias y por último los sinónimos de estos conceptos que redirigen a las instancias. Toda esta información que forma parte de los nodos de la taxonomía se utilizará más adelante, en el proceso de anotación de contexto de las reviews, para *mapear* palabras de las reviews con alguna de las categorías contextuales de la taxonomía.

### 4.1.3 Edición manual de la taxonomía

Una vez se ha construido de manera automática la taxonomía extrayendo las distintas entidades semánticas (categorías e instancias) de DBpedia, se puede editar de manera manual para ir expandiendo (mediante nuevas *queries* realizadas sobre las entidades de DBpedia) y añadiendo o eliminando las distintas categorías (y subcategorías) de la propia taxonomía. Este proceso es necesario ya que una vez extraídas las distintas categorías de DBpedia, es posible que se hayan extraído entidades que no representen categorías contextuales, por lo que no deben formar parte de la taxonomía de contexto.

En la siguiente figura se muestra una captura de la herramienta de construcción y edición de la taxonomía:

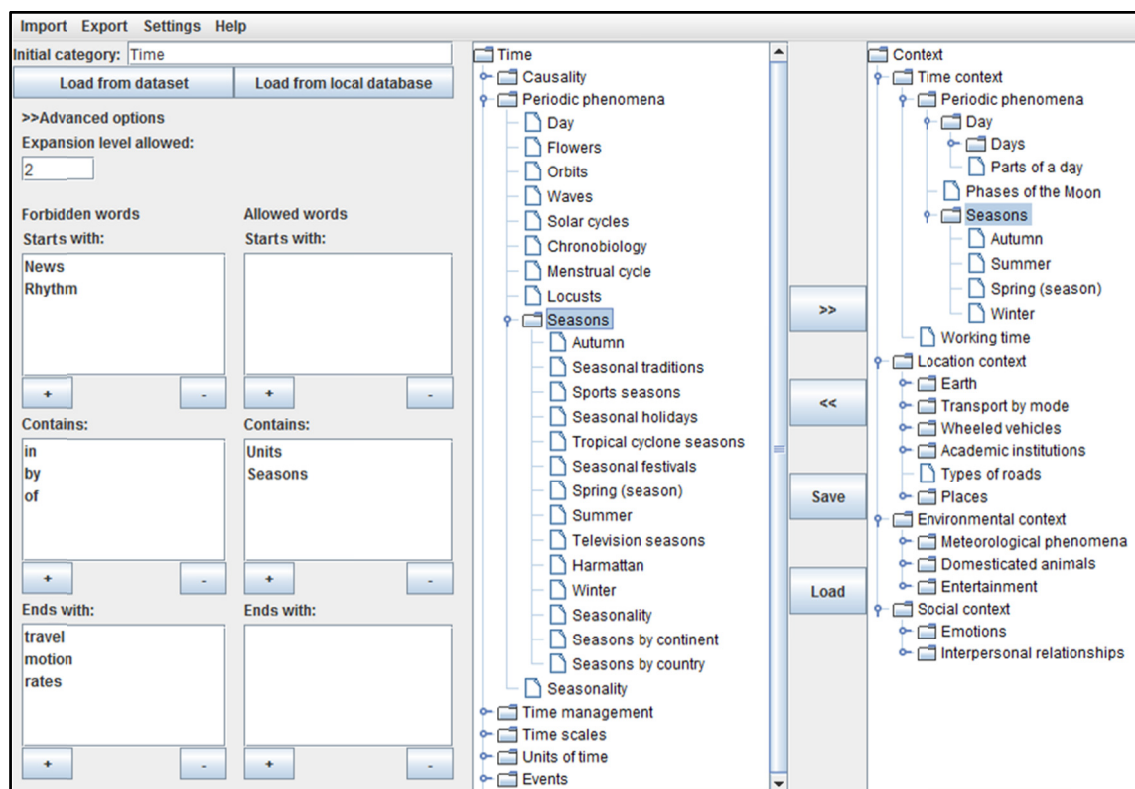


Figura 4.1 Herramienta de construcción y edición de la taxonomía de contexto.

Como puede observarse en la figura, la herramienta permite definir una serie de criterios a la hora de explorar las distintas entidades de DBpedia. Estos criterios se aplican sobre los nombres de las categorías para ver si éstas los satisfacen y deben explorarse o si por el contrario no los satisfacen y deben ser rechazadas. Por otro lado, también se puede establecer el nivel máximo de profundidad hasta el cual se quiere ir expandiendo las distintas entidades extraídas.

Se pueden establecer patrones sintácticos de tipos ‘empieza por’, ‘contiene’ y ‘termina con’ para aplicar sobre los nombres de categoría, con el fin de expandir o descartar aquellas categorías que respectivamente ‘empiecen por...’, ‘contengan...’ o ‘terminen con...’ cierto patrón o secuencia de letras. En la figura anterior se puede ver como se ha expandido la categoría ‘*Seasons*’ ya que se satisface la regla de expandir categorías que contengan la palabra ‘*Seasons*’.

Recordar que esta edición manual se hace necesaria para configurar la taxonomía contextual definitiva, ya que no todas las entidades extraídas de DBpedia de manera automática representan valores contextuales, por tanto, gracias a esos patrones sintácticos y a un continuo refinado de la taxonomía, se ha podido establecer la taxonomía definitiva para emplear en este trabajo.

Cabe destacar que este procedimiento de creación automática y edición manual asistida por la herramienta, es genérico y no se ve limitado a una taxonomía de contexto. De este modo, una vez seleccionadas las categorías raíz adecuadas, se podría emplear para generar taxonomías de otros dominios y aplicaciones, no relacionadas con aspectos contextuales. De este modo, el procedimiento propuesto y herramienta desarrollada suponen una contribución interesante a la literatura científica sobre creación automática de taxonomías.

#### 4.1.4 Taxonomía contextual final

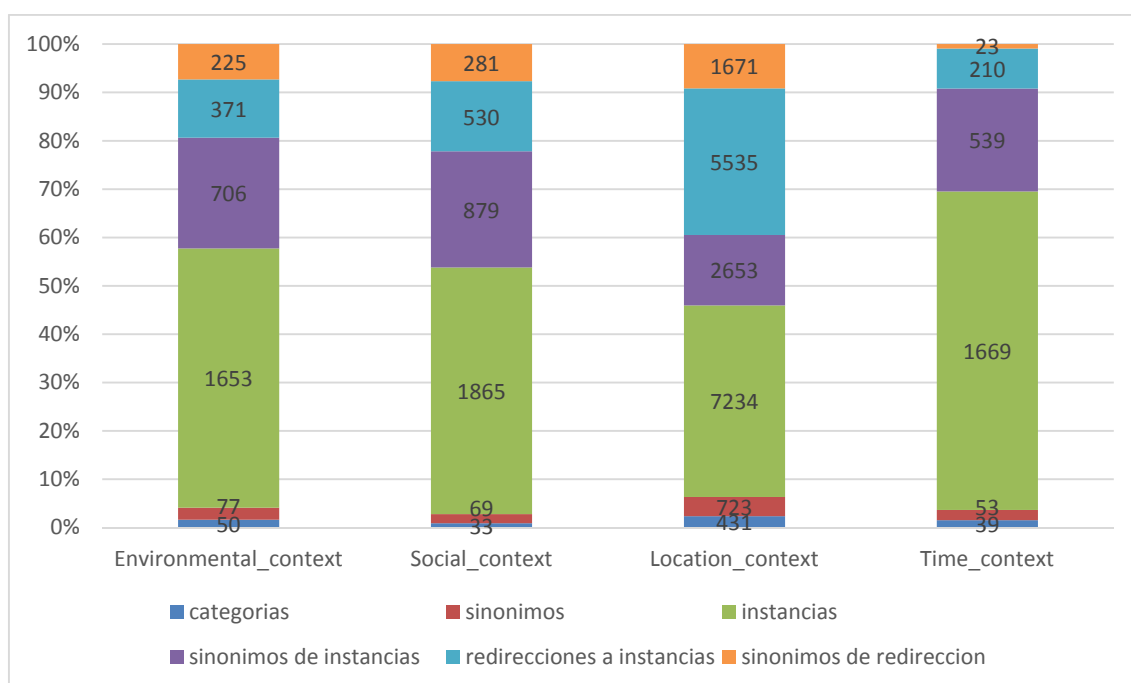
Una vez terminado el proceso de extracción automática de categorías semánticas de manera iterativa a través de DBpedia, la inclusión de información adicional a dichas categorías como pueden ser los sinónimos de la categoría y de las instancias, y la edición manual a través de la herramienta, se tiene lo que se ha considerado como la taxonomía genérica de contexto a explotar para anotar reviews y generar recomendaciones.

En total, la taxonomía cuenta con 554 categorías contextuales (considerando el noto raíz ‘Contexto’) repartidas entre las cuatro dimensiones consideradas, i.e., ‘*Time\_context*’, ‘*Location\_context*’, ‘*Environmental\_context*’ y ‘*Social\_context*’. En cuanto a la información adicional añadida a la taxonomía, ésta se compone de 922 sinónimos de dichas categorías, 12.421 instancias extraídas de DBpedia, 4.777 sinónimos para las instancias extraídas, 6.646 conceptos que redirigen a instancias de DBpedia y 2.200 sinónimos de dichos conceptos que redirigen a instancias. En la

siguiente tabla se muestra esta información estadística de la taxonomía separada según la dimensión contextual.

	Categorías contextuales	Sinónimos WordNet	Instancias DBpedia	Sinónimos instancias	Redirección a instancia	Sinónimo de redirección
<i>Time_context</i>	39	53	1.669	539	210	23
<i>Location_context</i>	431	723	7.234	2.653	5.535	1.671
<i>Environmental_context</i>	50	77	1.653	706	371	225
<i>Social_context</i>	33	69	1.865	879	530	281
<i>Total</i>	553	922	12.421	4.777	6.646	2.200

**Tabla 4.3** Estadísticas de la taxonomía genérica de contexto construida.

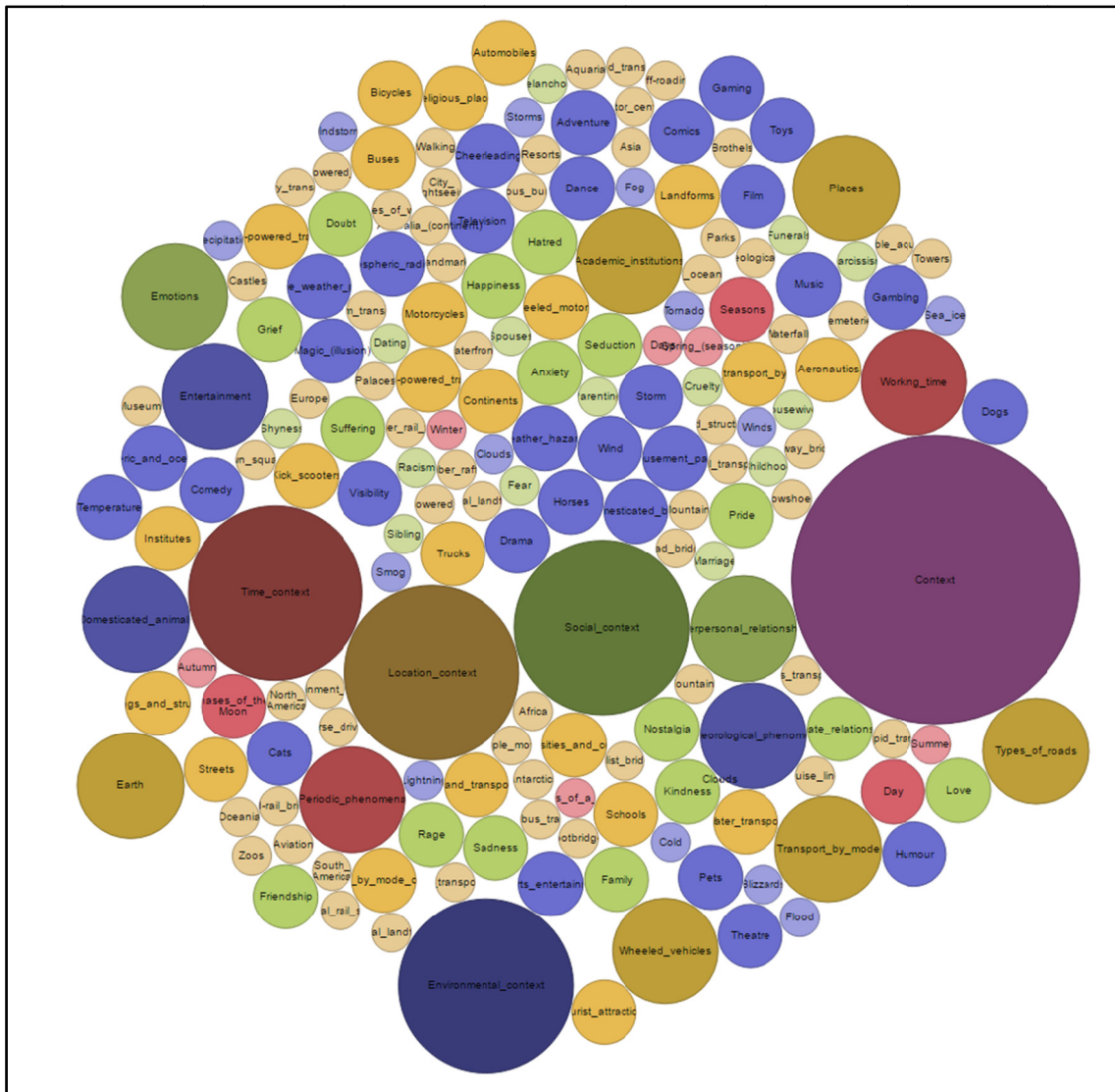


**Figura 4.2** Composición de la información adicional de la taxonomía.

Como puede observarse, la mayor parte de la información adicional asociada a la taxonomía está representada por instancias extraídas de DBpedia, y como es lógico debido al alto número de éstas, es seguido por los sinónimos de esas instancias y por los conceptos que redirigen las instancias. También se puede ver como la gran mayoría de las categorías contextuales de la taxonomía se encuentran bajo la dimensión '*Location\_context*', lo cual se debe a que dicha dimensión pretende encuadrar las posibles localizaciones de hechos y acciones; así, algunas de sus categorías cuentan con numerosas subcategorías, como puede ser la categoría '*Earth*' que agrupa los distintos países separados por continentes, además de la categoría '*Places*' que contiene

numerosas categorías contextuales para abarcar los distintos lugares donde podría tener lugar un hecho o acción contextualizada.

En la siguiente figura se muestran las distintas categorías contextuales existentes en la taxonomía hasta un nivel 4 de profundidad.



**Figura 4.3** Composición de la taxonomía hasta profundidad cuatro.

En la figura anterior, las distintas dimensiones contextuales de la taxonomía están representadas por colores similares, y un mayor tamaño representa un menor nivel dentro de la taxonomía, mientras que tamaños más pequeños representan niveles más profundos hasta llegar al nivel 4 que es el máximo nivel de profundidad representado en la figura.

## 4.2 Anotaciones contextuales en evaluaciones textuales

En esta sección se explica el proceso de anotación contextual de las reviews textuales de los usuarios. Para llevar a cabo esta anotación se emplea la taxonomía contextual que fue presentada en la sección anterior, y se aplican técnicas de procesamiento de lenguaje natural para mapear esas categorías, instancias y sinónimos contextuales existentes en la taxonomía con palabras usadas en las reviews escritas por los usuarios, de forma que se pueda etiquetar cada review con sus distintas categorías contextuales de manera automática.

El primer paso para llevar a cabo esta anotación es seleccionar dentro de todas las palabras de la review, aquellas que puedan ser candidatas como entidades contextuales, para después, en un segundo paso, intentar emparejar dichas palabras con las categorías de contexto contempladas dentro de la taxonomía genérica.

### 4.2.1 Procesamiento del lenguaje natural sobre evaluaciones textuales

Debido a la gran cantidad de palabras que se pueden encontrar dentro de las reviews de los usuarios, es necesario intentar distinguir cuales de ellas potencialmente pueden representar alguna categoría o instancia de contexto, ya que, si se revisaran todas las palabras de cada review se realizarían muchas anotaciones incorrectas, debido a la ambigüedad de las mismas, que provocaría que aunque dichas palabras puedan ser emparejadas con categorías de la taxonomía, no referenciarían al contexto en el que el usuario utilizó el ítem evaluado, sino a la propia descripción del ítem, o incluso a aspectos ajenos al propio ítem.

Más específicamente, analizando resultados previos, se comprobó que en gran parte de las reviews, los usuarios se dedican a describir los ítems. Por ejemplo, dentro de la frase *‘this film is about two friends that...’*, la palabra *‘friends’* puede encontrarse dentro de una categoría de la taxonomía, sin embargo, la frase no está diciendo que el usuario viera esa película con sus amigos (lo que si se consideraría contexto), sino simplemente está describiendo el ítem.

Por ello, el primer paso es identificar aquellas frases que puedan contener información contextual. En este trabajo esas frases se han establecido como aquellas en las que el autor describe una experiencia personal o expresa una opinión. Por ejemplo, en la frase *‘I watched this movie with my friends at home’*, el autor está expresando una experiencia personal, y las palabras *‘friends’* y *‘home’* representan contexto, y deberían ser anotadas dentro de las categorías contextuales *‘social’* y *‘localización’* respectivamente.



Para encontrar estas frases, se ha utilizado la librería *Stanford CoreNLP* (Manning et. al, 2014), la cual contiene una serie de funciones y utilidades para procesamiento de lenguaje natural, entre las cuales se encuentra la capacidad para extraer el *árbol sintáctico* de una frase.

De este modo, obteniendo y procesando los árboles sintácticos de las distintas frases que forman parte de las reviews, y haciendo uso del *Part-of-Speech (POS) tagger* – funcionalidad que permite conocer la categoría gramatical de las palabras– se han buscado aquellas frases escritas en primera persona del singular o plural, es decir, aquellas frases donde el sujeto es el pronombre ‘*I*’ o ‘*we*’; aquellas frases donde se emplea algún posesivo de primera persona como puede ser ‘*my*’, ‘*mine*’, ‘*our*’ o ‘*ours*’; y aquellas frases que emplean alguna preposición como ‘*in*’ o ‘*for*’ que vayan seguidas de un nombre, ya que pueden identificar contexto.

Una vez identificadas dichas frases potenciales de contexto, es necesario determinar qué palabras son las candidatas para representar alguna categoría de contexto. Según se ha podido observar, las palabras que emplean los usuarios para expresar contexto son **nombres** (e.g., *friends*, *Monday*, *cinema*, etc.) o determinados **adjetivos** (e.g., *sunny*, *happy*, *nervous*, etc.), por lo que de nuevo, empleando *CoreNLP* se han obtenido esas palabras potenciales de contexto, las cuales se intentan emparejar con alguna de las categorías de contexto existentes en la taxonomía.

#### 4.2.2 Reconocimiento de categorías e instancias contextuales en evaluaciones textuales

Una vez identificadas las palabras candidatas que son susceptibles de contener información contextual, se lleva a cabo un proceso para emparejar dichas palabras con las categorías contextuales que se encuentran en la taxonomía, para lo cual se emplea *Apache Lucene*<sup>13</sup>, la cual es una librería Java que permite realizar búsquedas sobre documentos de texto de una manera bastante eficiente.

Por tanto, el primer paso para poder utilizar la librería Apache Lucene, es construir un índice de “documentos” sobre los que buscar las palabras de contexto potenciales. Cada uno de estos documentos representa una de las categorías de contexto que se encuentra en la taxonomía, y está formado, además de por la propia categoría de contexto, por las distintas instancias y sinónimos que se habían asociado a dicha categoría.

Una vez construido este índice, al buscar una palabra candidata dentro de él, se obtendrá un ranking con los distintos documentos donde se encuentra esa palabra, que en definitiva no es otra cosa sino un ranking de las distintas categorías contextuales a las que puede pertenecer dicha palabra.

---

<sup>13</sup> Apache Lucene, librería para recuperación de información, <https://lucene.apache.org>

Antes de realizar la búsqueda dentro del índice Lucene, se lleva a cabo un proceso de *lematización* de la palabra candidata, para así poder lidiar con singulares y plurales, o con posibles errores morfológicos existentes dentro de la review, ya que, al ser contenido generado por los usuarios, pueden existir faltas ortográficas.

Debido a este proceso de lematización, antes de asignar la palabra candidata a alguna de las categorías contextuales (los documentos extraídos del índice Lucene) es necesario calcular el nivel de similitud entre la palabra candidata y aquellas palabras devueltas por el índice Lucene que se encontraban dentro de los documentos, para así quedarnos únicamente con aquellas búsquedas que resulten más efectivas. Así, aquellas que tengan un nivel de similitud mayor, y que pertenezcan a la categoría de *matching* más alta (ver Tabla 4.4.), se asignarán a la palabra candidata.

Nivel de <i>matching</i>	
1 – <i>Coincidencia exacta</i>	Coincidencia exacta entre la categoría contextual y la palabra
2 – <i>Sinónimo</i>	Coincidencia entre un sinónimo de la categoría (obtenido de WordNet) y la palabra.
3 – <i>Instancia</i>	Coincidencia entre una instancia de la categoría (obtenida de DBpedia) y la palabra.
4 – <i>Sinónimo de instancia</i>	Coincidencia entre un sinónimo de una instancia de la categoría y la palabra.
5 – <i>Redirección a instancia</i>	Coincidencia entre la palabra y un concepto de DBpedia que redirige a una instancia de la categoría.
6 – <i>Sinónimo de redirección</i>	Coincidencia entre la palabra y un sinónimo de un concepto de DBpedia que redirige a una instancia de la categoría.

**Tabla 4.4** Niveles de matching entre palabras de las reviews y las categorías de la taxonomía de contexto.

Para establecer el nivel de similitud entre dos palabras se ha empleado la distancia de Damerau-Levenshtein (Dameraun, 1964 y Levenshtein, 1966), la cual permite calcular la distancia de edición entre dos palabras, entendiendo por distancia de edición el mínimo número de operaciones requeridas para transformar una palabra en otra. Dichas operaciones hacen referencia al número de inserciones, eliminaciones, sustituciones o trasposiciones de caracteres que hay que aplicar para transformar la palabra. Esa distancia, junto con la longitud de las palabras, permite calcular la similitud de ambas palabras, empleando la siguiente formula:

$$sim(w_1, w_2) = \frac{L - 2 * D}{L}$$

donde  $D$  representa la distancia Damerau-Levenshtein entre las palabras  $w_1$  y  $w_2$ , y  $L$  se calcula como  $L = \min(longitud(w_1), longitud(w_2))$ .

El resultado de todo el proceso explicado en esta sección para cada review es un conjunto de anotaciones de contexto que están compuestas de la palabra original de la review, la categoría de la taxonomía de contexto asociada a esa palabra, el nombre de categoría/instancia/sinónimo mapeada, y el nivel de *matching*. Explotando tales anotaciones, en este trabajo se exploran y evalúan técnicas de recomendación contextualizada, que se describen en la siguiente sección.

## 4.3 Métodos de recomendación contextualizada

En esta sección se detallan los métodos de recomendación empleados sobre las anotaciones de contexto generadas según se ha explicado en las secciones anteriores. En primer lugar, se explican cuales son aquellos métodos de recomendación base empleados como referencia (*baseline*) que no tienen en cuenta ningún aspecto del contexto, únicamente emplean los valores de los ratings del conjunto de usuarios e ítems. Estos métodos *baseline*, servirán para comparar el resto de métodos que sí tienen en cuenta los valores de las anotaciones contextuales para así medir si los métodos que tienen en cuenta los valores contextuales anotados son más acertados que los métodos base que no tienen en cuenta los contextos anotados. Estos métodos se encuentran dentro del motor *CARSKit*<sup>14</sup> (Zheng et al., 2015), una librería Java open-source diseñada para recomendación contextualizada.

### 4.3.1 Métodos base de referencia

En los experimentos llevados a cabo con las anotaciones de contexto generadas (que serán detallados en la siguiente sección) se han empleado los siguientes métodos base (*baselines*) como referencia:

- **GlobalAverage:** este método de recomendación asigna al rating predicho el valor correspondiente a la media global calculada sobre todos los ratings del dataset.
- **UserAverage:** este método asigna como valor predicho de rating la media correspondiente al usuario calculada a partir de todos sus ratings. En caso de que sea un usuario nuevo, se asigna la media global de todo el dataset.
- **ItemAverage:** este método asigna como valor predicho la media correspondiente a los ratings asociados al ítem en concreto. En caso de que el ítem no tenga ningún rating asociado (en la fase de entrenamiento), se asigna la media global de todo el dataset.

---

<sup>14</sup> CARSKit, motor para recomendación contextualizada, <https://github.com/irecsys/CARSKit>

- **UserKNN** (Resnick et al., 1994): método de filtrado colaborativo que emplea kNN (algoritmo de k-vecinos próximos) basado en usuario, que como se ha explicado en la Sección 2.1.2, estima el rating de un ítem empleando aquellos  $k$  usuarios (vecinos) similares al usuario objetivo  $u$ , agregando el valor de rating asignado por esos usuarios al ítem objetivo  $i$ .
- **ItemKNN** (Sarwar et al., 2001): método de filtrado colaborativo que emplea kNN basado en ítem, que como se explicó en la Sección 2.1.2, estima el rating del ítem objetivo  $i$  empleando los ratings que el usuario objetivo  $u$  ha realizado sobre los  $k$  ítems (vecinos) más similares al ítem objetivo.
- **BiasedMF** (Koren et al, 2009): método basado en factorización de matrices, el cual consiste en caracterizar los usuarios e ítems por vectores o factores latentes, los cuales son inferidos a partir de los patrones existentes en los ratings de los ítems. En su método más básico, los usuarios e ítems se mapean en un espacio de factores latentes de dimensionalidad  $f$ , donde cada ítem  $i$  es asociado con un vector  $q_i \in \mathbb{R}^f$ , y cada usuario  $u$  es asociado con un vector  $p_u \in \mathbb{R}^f$ , de forma que la interacción entre usuario e ítem se puede medir a través del producto escalar de ambos vectores, lo cual sirve para aproximar el rating que el usuario  $u$  ha dado al ítem  $i$ , de la siguiente forma  $\hat{r}_{ui} = q_i^T p_u$ .

La fórmula anterior trata de capturar las interacciones entre usuarios e ítems para producir los diferentes valores de rating, sin embargo, las variaciones en los ratings normalmente están asociadas directamente con los usuarios o ítems de manera independiente a la interacción entre ellos. Estas variaciones son conocidas como sesgos (*biases*), y sirven para representar hechos como por ejemplo que ciertos usuarios tienen a dar puntuaciones más altas sin importar el ítem que estén valorando, o que ciertos ítems tienen a recibir ratings más altos por parte de todos los usuarios en general. De esta forma el sesgo introducido en los ratings  $r_{ui}$  se define como  $b_{ui} = \mu + b_i + b_u$  donde  $\mu$  representa la puntuación media global de todos los ratings,  $b_u$  y  $b_i$  representan las desviaciones del usuario  $u$  y del ítem  $i$  de esa media global. Por lo que, la fórmula para estimar el rating queda de la siguiente manera:

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u$$

De esta forma, el rating estimado se descompone en cuatro componentes, la media global, el sesgo del ítem, el sesgo del usuario y la interacción usuario-ítem. Por lo que, para llevar a cabo la fase de entrenamiento del modelo, se minimiza el error cuadrático de la siguiente forma:

$$\min_{p^*, q^*, b^*} \sum_{(u,i) \in K} (r_{ui} - \mu - b_u - b_i - q_i^T p_u)^2 + \lambda(\|p_u\|^2 + \|q_i\|^2 + b_u^2 + b_i^2)$$

Como este método de aprendizaje tiende a terminar sobre-ajustándose a los ratings observados, lo que se conoce como sobre entrenamiento u *overfitting*. Con el fin de evitarlo, se emplea un factor de regularización  $\lambda$ , y se realizan un número máximo de iteraciones de la fase de entrenamiento, para conseguir así un modelo más genérico y evitar el *overfitting*.

### 4.3.2 Métodos conscientes del contexto

Además de los métodos de referencia explicados anteriormente, se han empleado algunos métodos de recomendación que son conscientes del contexto, para así poder llevar a cabo una comparativa y ver si las anotaciones de contexto generadas resultan beneficiosas a la hora de realizar la recomendación. Los métodos utilizados son la factorización de matrices con información contextual o **CAMF**, la técnica de **Item Splitting**, y la técnica de **UI Splitting** (técnica similar a *Item Splitting* solo que aplicada también a los usuarios).

El primero de los métodos analizados es la factorización de matrices con información contextual de los ratings, conocida como **Context-aware Matrix Factorization**, **CAMF** (Baltrunas et al. 2011). Este método representa una extensión del método tradicional de factorización de matrices que incluye nuevos parámetros para representar la interacción de las condiciones contextuales con los usuarios e ítems. De la misma forma que el método de factorización de matrices tradicional, se emplea un espacio de factores latentes de dimensionalidad  $f$ , donde cada ítem  $i$  es asociado a un vector  $q_i \in \mathbb{R}^f$ , y cada usuario  $u$  es asociado a un vector  $p_u \in \mathbb{R}^f$ , quedando la fórmula para estimar el rating de la siguiente manera:

$$\hat{r}_{ui} = \bar{r} + b_u + q_i^T p_u + \sum_{j=1}^k B_{ijc_j}$$

Donde  $\bar{r}$  representa el valor medio del ítem,  $b_u$  representa la desviación del usuario  $u$  al igual que ocurría en el modelo tradicional, y  $B_{ijc_j}$  son los parámetros que modelan la interacción de las condiciones contextuales con los ítems (o con los usuarios como veremos más adelante), donde  $k$  representa el número de factores o dimensiones contextuales.

Dentro de este método de factorización de matrices con información contextual se pueden extraer distintos modelos según el nivel de complejidad que se quiera asignar a la componente  $B_{ijc_j}$  encargada de modelar la interacción de las condiciones contextuales con los ítems o usuarios. De esta forma, se han empleado los 4 modelos posibles:

- **CAMF\_C**: es el modelo más simple de todos, asumiendo que los factores o dimensiones contextuales tienen una influencia uniforme sobre los todos los ratings, sin diferenciar por ítem, de esta manera solo existe un parámetro  $B_{jc_j}$

por cada factor o dimensión contextual, o lo que es lo mismo que  $B_{ijc_j} = B_{fjc_j}$  para todo par de ítems  $i$  y  $f$ . Por lo que la fórmula quedaría de la siguiente manera:

$$\hat{r}_{ui} = \bar{r} + b_u + q_i^T p_u + \sum_{j=1}^k B_{jc_j}$$

- **CAMF\_CI**: este modelo asume que cada factor contextual influye de manera diferente a cada ítem, por tanto, existe un parámetro  $B_{ijc_j}$  por cada dimensión contextual e ítem  $i$ . Por lo que la fórmula para estimar el rating se corresponde con la fórmula original:

$$\hat{r}_{ui} = \bar{r} + b_u + q_i^T p_u + \sum_{j=1}^k B_{ijc_j}$$

- **CAMF\_CU**: este modelo es similar al modelo anterior, solo que desde el punto de vista de los usuarios. Desde este punto de vista se asume que cada factor contextual influye de manera diferente a los usuarios, por lo que existen tantos parámetros  $B_{ujc_j}$  por cada dimensión contextual y usuario  $u$ .

$$\hat{r}_{ui} = \bar{r} + b_u + q_i^T p_u + \sum_{j=1}^k B_{ujc_j}$$

- **CAMF\_CUCI**: este modelo combina las dos aproximaciones anteriores, asumiendo que cada factor contextual influye de manera diferente tanto en los ítems como en los usuarios. De esta forma, se incluye un parámetro  $B_{ijc_j}$  por cada dimensión contextual e ítem, y un parámetro  $B_{ujc_j}$  por cada dimensión contextual y usuario. Por tanto, la fórmula para estimar el rating utilizando este modelo sería la siguiente:

$$\hat{r}_{ui} = \bar{r} + b_u + q_i^T p_u + \sum_{j=1}^k B_{ijc_j} + \sum_{j=1}^k B_{ujc_j}$$

CAMF es un ejemplo de método de recomendación contextualizada basado en modelo, puesto que el modelo aprendido incorpora directamente información de contexto. Los demás métodos utilizados corresponden a métodos de pre-filtrado contextual, pues en ellos se realiza un pre-procesamiento de los datos según las condiciones contextuales, para luego aplicar métodos de recomendación tradicionales que no tienen en cuenta el contexto sobre los datos ya filtrados.

El primero de ellos es el denominado *Item Splitting* (Baltrunas & Ricci, 2009), donde se emplea la idea de que los ratings de los ítems pueden ser divididos en dos subconjuntos de ratings de acuerdo al valor de un determinado factor contextual. De

forma que el ítem queda dividido en dos pseudo-items cada uno asignado al subconjunto de ratings correspondientes que se integran de nuevo junto con el resto de ítems del sistema.

Esta división se realiza solo si existe evidencia estadística de que ambos subconjuntos de ratings son independientes, lo que implica que el valor contextual por el que se han separado influye a la hora de generar las recomendaciones. Para determinar esto se emplean ‘criterios de impureza’ (*impurity criteria*) que eligen aquellos factores contextuales que maximicen cierta métrica estadística y en caso de superar cierto umbral, se lleva a cabo la separación de los ratings según el valor contextual. El algoritmo para llevar a cabo este método es el siguiente:

**Entrada:** Ratings del dataset, criterio de impureza  $t$ , umbral  $d$

**Salida:** Dataset con los ratings modificados

**Para cada** ítem  $i$

**Para cada** factor contextual  $C_j \in C_1 \dots C_k$

**Para cada** valor contextual  $c \in C_j$

            Generar los ratings  $r_{i_c}$  y  $r_{i_{\bar{c}}}$

            Calcular el criterio de impureza  $t(i, c)$

$c_{max} \leftarrow \arg \max_c \{t(i, c)\}$

**Si**  $t(i, c_{max}) > d$  **entonces**

        Modificar dataset reemplazando ítem  $i$  por  $i_c$  e  $i_{\bar{c}}$

En este caso, se ha utilizado el criterio de impureza correspondiente a la prueba  $t$  de Student o  $t$ -Test a que exista diferencia significativa entre las medias de ambos subconjuntos de ratings de un determinado ítem  $i$  bajo una determinada condición contextual  $c$ . Este criterio de impureza se denomina  $t_{mean}$  y su fórmula es la siguiente:

$$t_{mean} = \left| \frac{\mu_{i_c} - \mu_{i_{\bar{c}}}}{\sqrt{S_{i_c}/n_{i_c} + S_{i_{\bar{c}}}/n_{i_{\bar{c}}}}} \right|$$

donde  $\mu_{i_c}$  representa la media de los ratings del ítem bajo el valor  $c$  de la condición contextual,  $\mu_{i_{\bar{c}}}$  es la media de los ratings del ítem bajo el resto de valores de la condición contextual distintos de  $c$ ,  $S_{i_c}$  y  $S_{i_{\bar{c}}}$  representan la varianza de los ratings del ítem  $i$  bajo el valor  $c$  de una determinada condición contextual y bajo el resto respectivamente, y por ultimo  $n_{i_c}$  y  $n_{i_{\bar{c}}}$  representan el número de ratings del ítem  $i$  bajo el valor  $c$  de la condición contextual, y bajo el resto de valores distintos de  $c$ . Bajo este criterio, cuando más grande es el valor de  $t$  mayor diferencia entre las medias existe, por lo que en caso de alcanzar un valor mayor que 4, estaríamos hablando de que se correspondería con el nivel 0.05 de significatividad estadística. En tal caso se entendería que ambos subconjuntos de ratings son independientes y el ítem será dividido en dos pseudo-items teniendo en cuenta el valor de la condición contextual.

Esta técnica puede modificarse para en lugar de dividir los ítems, dividir los usuarios en pseudo-usuarios tomando los subconjuntos de ratings realizados por los usuarios bajo una determinada condición contextual, y de la misma forma que antes, en caso de que exista evidencia estadística de que ambos subconjuntos son diferentes, llevar a cabo la separación. Esta técnica se conoce como *User Splitting*, de forma que combinándola junto con la técnica anterior de *Item Splitting*, aparece la técnica conocida como ***UI Splitting*** (Zheng et al. 2014), la cual modifica los datos iniciales para dividir tanto los ítems como los usuarios según los valores de las condiciones contextuales, siguiendo el mismo procedimiento que se empleaba en el *Item Splitting* original, sólo que aplicado también a los usuarios.



## Capítulo 5

# Experimentos

En este capítulo se explican los experimentos llevados a cabo. En primer lugar, se detallan los datos utilizados; en concreto, se han empleado 100.000 reviews de un *dataset* de reviews de Amazon para tres tipos de ítems que se emplean habitualmente en los trabajos de recomendación: películas, música y libros. Estas reviews han sido anotadas de contexto de manera automática empleando la taxonomía genérica de contexto que se ha explicado previamente en la Sección 4.1. Posteriormente se detalla el proceso de evaluación manual llevado a cabo sobre esas anotaciones de contexto empleando para ello una herramienta de validación que además de permitir evaluar las anotaciones, permite incluir nuevas anotaciones que el algoritmo de anotación no haya encontrado. Por último, se presentan y discuten los resultados obtenidos empleando distintos algoritmos de recomendación conscientes del contexto sobre los datos anotados.

### 5.1 Repositorio de evaluaciones textuales

Para llevar a cabo los experimentos se ha utilizado parte del dataset de reviews de Amazon<sup>15</sup> publicado por McAuley y Leskovec (2013), el cual incluye reviews extraídas del sitio de comercio electrónico Amazon.com durante un periodo de 18 años, incluyendo en torno a 35 millones de reviews (marzo 2013). Cada una de las reviews incluye información sobre el producto (ítem) evaluado, el usuario que evalúa, el rating (en escala 1-5) asignado por el usuario al producto, y la review del usuario en formato de texto plano.

---

<sup>15</sup> Dataset de reviews de Amazon, <https://snap.stanford.edu/data/web-Amazon.html>

A modo de descripción general, en la siguiente tabla se ofrecen algunos datos estadísticos sobre el contenido total de dicho dataset:

<i>Número de reviews</i>	34.686.770
<i>Número de usuarios</i>	6.643.669
<i>Número de ítems</i>	2.441.053
<i>Usuarios con más de 50 reviews</i>	56.772
<i>Mediana palabras por review</i>	82
<i>Espacio de tiempo</i>	Junio 1995 – marzo 2013

**Tabla 5.1** Descripción general del dataset de reviews de Amazon.

A su vez, las reviews se encuentran distribuidas por categorías, dependiendo de los ítems a los que se hagan referencia. Para este trabajo se han seleccionado las reviews de tres dominios que se emplean habitualmente en el mundo de la recomendación, como son las películas, los libros y la música.

En la siguiente tabla se muestra el número de reviews, usuarios e ítems que se encuentran en cada uno de los tres dominios considerados.

	<b>Películas</b>	<b>Música</b>	<b>Libros</b>
<i>Número de reviews</i>	728.220	105.607	1.209.093
<i>Número de usuarios</i>	420.622	78.373	816.917
<i>Número de ítems</i>	3.421	6.492	16.358

**Tabla 5.2** Números totales de reviews, usuarios e ítems del dataset de Amazon en cada uno de los tres dominios considerados.

Un ejemplo del formato de review dentro del dataset es el siguiente:

```
{
  "reviewerID": "A2SUAM1J3GNN3B",
  "asin": "0000013714",
  "reviewerName": "J. McDonald",
  "helpful": [2, 3],
  "reviewText": "I bought this for my husband who plays the piano. He is having
a wonderful time playing these old hymns. The music is at times hard to read
because we think the book was published for singing from more than playing
from. Great purchase though!",
  "overall": 5.0,
  "summary": "Heavenly Highway Hymns",
  "unixReviewTime": 1252800000,
  "reviewTime": "09 13, 2009"
}
```

En él se puede observar los campos relativos al usuario que hizo la review (*reviewerID*), el identificador del ítem (*asin*), el texto de la review que escribió el usuario (*reviewText*) y el asunto de dicho texto (*summary*), así como el valor de rating que el usuario asignó a dicho ítem (*overall*), y el momento del tiempo en el que realizó dicha review (*unixReviewTime* y *reviewTime*). Para este trabajo, el nombre del usuario (*reviewerName*) y el valor que indica si el comentario es útil (*helpful*) no se han utilizado.

Debido a la gran cantidad de reviews del dataset para los tres dominios considerados, se han escogido únicamente las 100.000 primeras reviews de cada dominio para ser anotadas con contexto de manera automática, y posteriormente utilizadas dentro de las tareas de recomendación.

En la siguiente tabla se muestra el número de reviews, usuarios e ítems empleados en los experimentos de este trabajo.

	<b>Películas</b>	<b>Música</b>	<b>Libros</b>
<i>Número de reviews</i>	100.000	100.000	100.000
<i>Número de usuarios</i>	77.030	74.131	87.899
<i>Número de ítems</i>	323	6.017	377

**Tabla 5.3** Números totales de reviews, usuarios e ítems empleados en los experimentos en cada uno de los tres dominios considerados.

Una vez seleccionadas las 100.000 primeras reviews de cada dominio, éstas se pasan por el proceso encargado de realizar la anotación automática de contexto. A continuación, se analiza a nivel general cómo quedan esas reviews, usuarios e ítems tras realizar el proceso de anotación.

En la siguientes tabla y figura, se muestra el número total de reviews disponibles para cada dominio, indicando cuantas de ellas tienen al menos una anotación de contexto, y cuantas de ellas no poseen ningún tipo de anotación.

<b>Dominio</b>	<b>Número de reviews</b>	<b>Con algún contexto anotado</b>	<b>Sin anotaciones de contexto</b>
<i>Películas</i>	100.000	29.893	70.107
<i>Música</i>	100.000	31.080	68.920
<i>Libros</i>	100.000	28.479	71.521
<i>Total</i>	300.000	89.452	210.548

**Tabla 5.4** Total de reviews con anotaciones y sin anotaciones de contexto.

No todas las reviews poseen alguna anotación de contexto, bien sea porque realmente carecen de información contextual, bien porque el algoritmo desarrollado no ha sido capaz de realizar anotaciones. Como se puede observar en la tabla, tras realizar la anotación contextual, se ha obtenido que aproximadamente un 30% del total de reviews tienen alguna anotación de contexto.

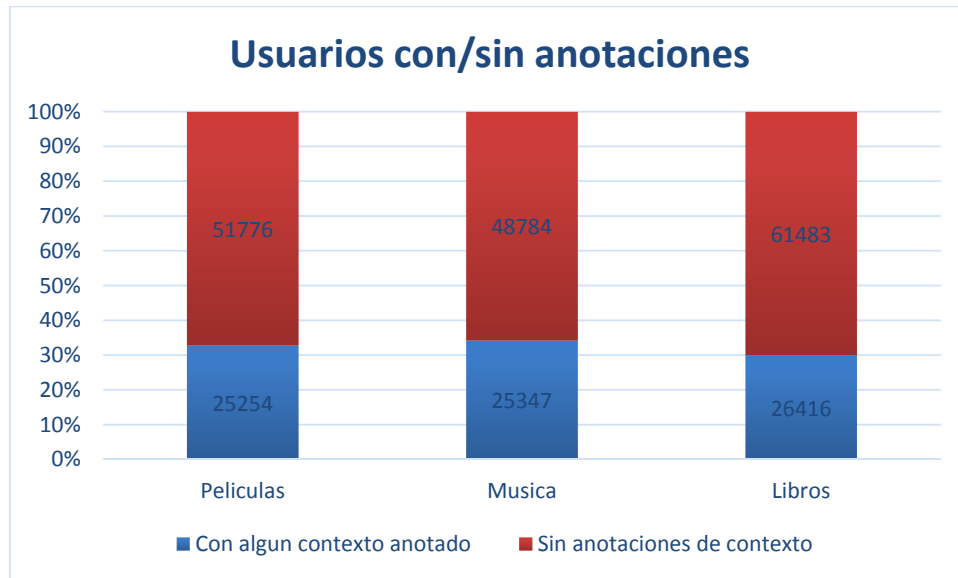


**Figura 5.1** Total de reviews con anotaciones y sin anotaciones de contexto.

En las siguientes tabla y figura, se muestra el número de usuarios totales disponibles para cada dominio, y cuántos de esos usuarios poseen alguna anotación de contexto y cuantos no tienen ninguna tras realizar el proceso automático de anotación.

Dominio	Número de usuarios	Con algún contexto anotado	Sin anotaciones de contexto
<i>Películas</i>	77.030	25.254	51.776
<i>Música</i>	74.131	25.347	48.784
<i>Libros</i>	87.899	26.416	61.483

**Tabla 5.5** Total de usuarios con anotaciones y sin anotaciones de contexto.



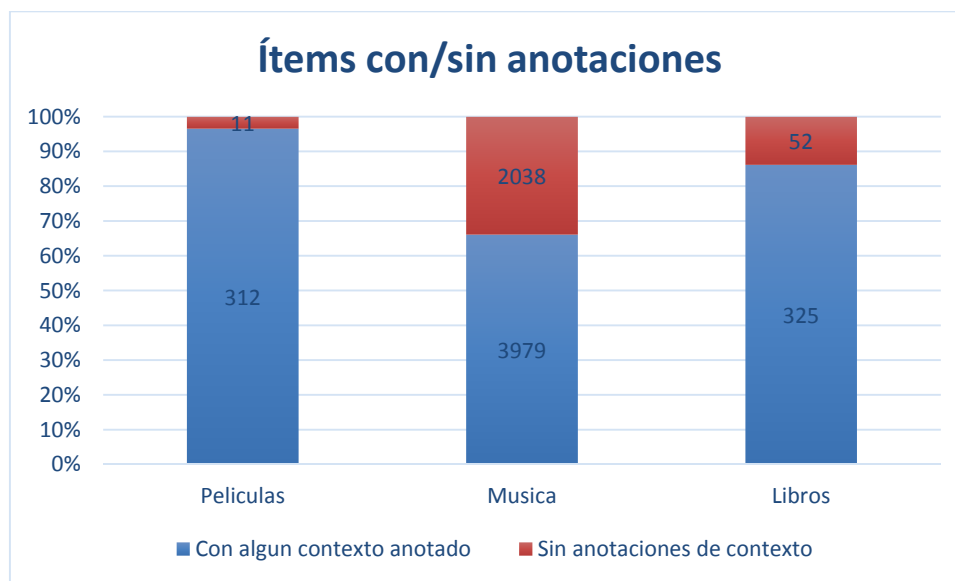
**Figura 5.2** Total de usuarios con anotaciones y sin anotaciones de contexto.

Como era de esperar, no todos los usuarios poseen anotaciones de contexto, y como puede observarse en la tabla anterior, aproximadamente sólo un 30% de los usuarios totales existentes para cada dominio poseen como mínimo alguna anotación de contexto en alguna de sus reviews.

Del mismo modo que los usuarios, se puede analizar el número total de ítems para ver qué porcentaje de ellos tiene al menos una anotación de contexto. En las siguientes tabla y figura se muestran los distintos ítems de cada dominio, indicando cuántos de ellos tienen alguna anotación de contexto, frente a cuantos no tienen ninguna. Como puede observarse, se ha conseguido identificar algún tipo de contexto para la gran mayoría de los ítems de cada uno de los tres dominios.

Dominio	Número de ítems	Con algún contexto anotado	Sin anotaciones de contexto
<i>Películas</i>	323	312	11
<i>Música</i>	6.017	3.979	2.038
<i>Libros</i>	377	325	52
<i>Total</i>	6.717	4.616	2.101

**Tabla 5.6** Total de ítems con anotaciones y sin anotaciones de contexto.



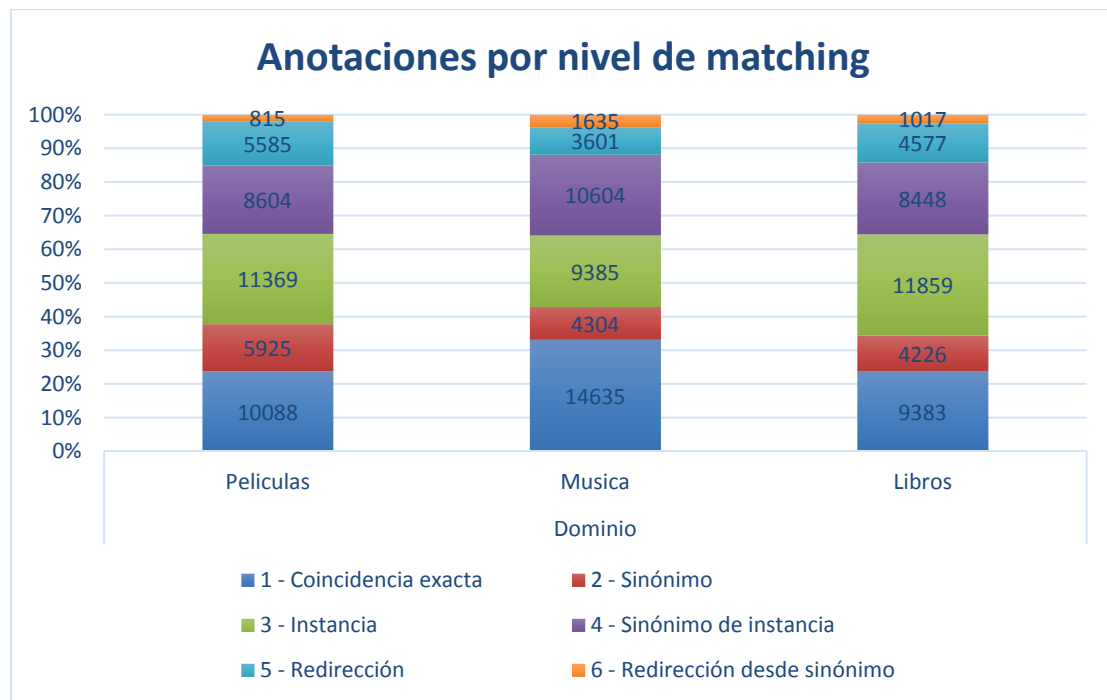
**Figura 5.3** Total de ítems con anotaciones y sin anotaciones de contexto.

Por último, si analizamos los distintos niveles de *matching* disponibles que se empleaban en el proceso de anotación contextual por cada dominio, uno se puede hacer una idea de qué emparejamientos se están produciendo con mayor frecuencia.

En las siguientes tabla y figura se muestran los totales de número de anotaciones de contexto realizadas en las distintas reviews de los tres dominios, distinguiendo el nivel de matching empleado en el proceso de anotación.

Nivel de matching	Películas	Música	Libros	Total
<i>Coincidencia exacta</i>	10.088	14.635	9.383	34.106
<i>Sinónimo</i>	5.925	4.304	4.226	14.455
<i>Instancia</i>	11.369	9.385	11.859	32.613
<i>Sinónimo de instancia</i>	8.604	10.604	8.448	27.656
<i>Redirección</i>	5.585	3.601	4.577	13.763
<i>Redirección desde sinónimo</i>	815	1.635	1.017	3.467
<i>Total anotaciones</i>	42.386	44.164	39.510	126.060

**Tabla 5.7** Números totales de anotaciones en cada dominio y nivel de matching.



**Figura 5.4** Niveles de matching empleados en la anotación de cada dominio.

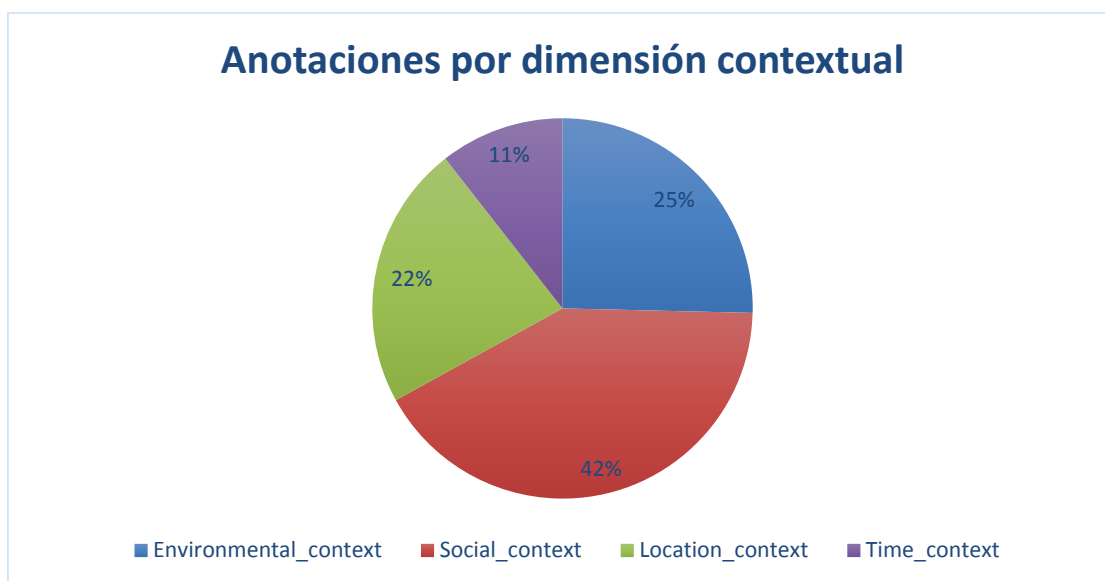
Como puede observarse, el mayor número de anotaciones se consigue mediante una coincidencia exacta de la palabra con la categoría contexto, o bien a través de las instancias obtenidas de DBpedia, o de los sinónimos de dichas instancias, mientras que los sinónimos obtenidos directamente de WordNet representan una menor parte dentro del total de anotaciones.

Por último, se ve como algunas de las anotaciones se han obtenido empleando las clases o sinónimos que redirigían a la propia categoría contextual, aunque representan un menor porcentaje del total de anotaciones, lo cual es acertado, ya que a través de este tipo de emparejamiento es probable que se esté realizando un mayor número de anotaciones incorrectas frente a los otros tipos, como pueden ser las propias instancias de DBpedia o los sinónimos de las categorías contextuales.

Por último, se han agrupado las distintas anotaciones según la dimensión contextual a la que pertenecen, ya sea localización, tiempo, entorno o social. En las siguientes tabla y figura se muestra dicha información. Como puede observarse, a pesar de que el total de anotaciones es de 126.060, 126 han sido anotadas como contexto en general, sin pertenecer a ninguna de las cuatro dimensiones de contexto, de ahí que el total de anotaciones pertenecientes a estas dimensiones sea de 125.934:

Dimensión contextual	Número de anotaciones	Porcentaje
<i>Environmental_context</i>	31.952	25,37 %
<i>Social_context</i>	52.416	41,62 %
<i>Location_context</i>	28.260	22,44 %
<i>Time_context</i>	13.306	10,57 %
<i>Total</i>	125.934	100 %

**Tabla 5.8** Anotaciones según su dimensión contextual.



**Figura 5.5** Anotaciones distribuidas por dimensión contextual.

Se puede ver como la dimensión contextual más presente es la correspondiente a al ‘contexto social’, en ella se agrupan, entre otros, contextos que indican con qué personas compartió el ítem el usuario (e.g., si se trata de una película, con quien la vio), o las distintas emociones o sentimientos que provocó el ítem al usuario.

En los siguientes apartados se analizan los resultados de cada dominio de manera particular, incluyendo además información acerca de las categorías contextuales más utilizadas en el proceso de anotación, lo que podría significar los contextos más representativos para dichos dominios.

### 5.1.1 Evaluaciones textuales de películas

En la siguiente tabla se detallan el número total de reviews, usuarios e ítems que se han anotado con al menos un contexto, y aquellos donde no se ha podido realizar ninguna anotación.



	<b>Total</b>	<b>Con contexto</b>	<b>Sin anotaciones</b>	<b>% anotación</b>
<i>Número de reviews</i>	100.000	29.893	70.107	29,89 %
<i>Número de usuarios</i>	77.030	25.254	51.776	32,78 %
<i>Número de ítems</i>	323	312	11	96,59 %

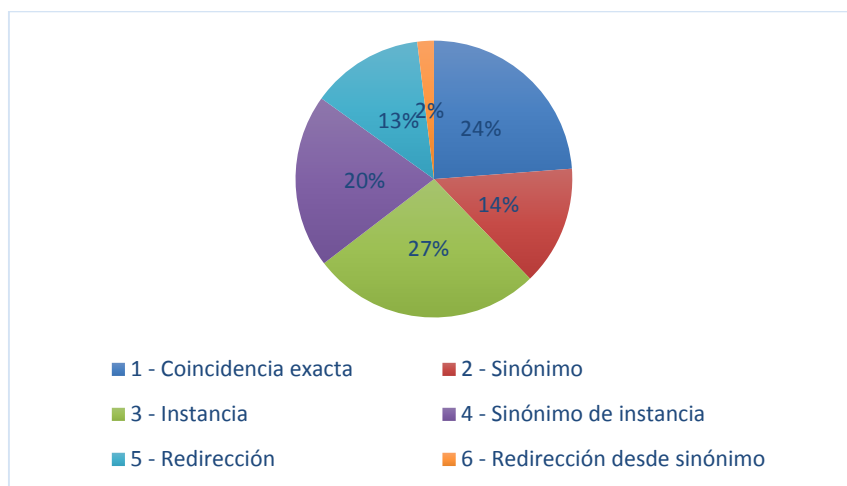
**Tabla 5.9** Reviews, usuarios e ítems con/sin anotaciones para el dominio de películas.

Como se puede observar, se ha conseguido anotar prácticamente el 30% de las reviews disponibles con alguna categoría contextual. Desde el punto de vista de los usuarios que realizaron estas reviews, aproximadamente un 33% de ellos cuenta con al menos una de sus reviews anotadas con contexto. Por último, revisando los distintos ítems en particular, se puede observar que se ha conseguido anotar prácticamente el total de los ítems, un 97%, con al menos alguna de sus reviews con contexto, esto representa el mayor porcentaje de anotación de ítems de entre los tres dominios.

En cuanto a los niveles de matching empleados, se puede ver como gran parte de los emparejamientos representan una coincidencia exacta, o bien una coincidencia con una instancia o un sinónimo de alguna instancia de DBpedia. Estos tres niveles de matching representan casi el 71 % de las anotaciones contextuales realizadas.

<b>Nivel de matching</b>	<b>Número</b>	<b>Porcentaje</b>
<i>Coincidencia exacta</i>	10.088	23,80 %
<i>Sinónimo</i>	5.925	13,98 %
<i>Instancia</i>	11.369	26,82 %
<i>Sinónimo de instancia</i>	8.604	20,30 %
<i>Redirección</i>	5.585	13,18 %
<i>Redirección desde sinónimo</i>	815	1,92 %
<i>Total anotaciones</i>	42.386	100 %

**Tabla 5.10** Niveles de matching empleados en las reviews de películas.



**Figura 5.6** Niveles de matching empleados en las reviews de películas.

Para el dominio de películas, en total se han realizado 42.386 anotaciones de contexto, empleando un total de 293 categorías contextuales de la taxonomía. En la siguiente tabla se incluyen los 15 contextos anotados con mayor frecuencia para este dominio, indicando también la dimensión contextual a la que pertenecen, es decir, la rama de nivel 1 de la taxonomía donde se encuentran y el porcentaje que representa ese número de anotaciones respecto al total de 42.386 anotaciones realizadas.

Categoría contextual	Dimensión contextual	Número anotaciones	Porcentaje respecto al total
<i>Family</i>	Social_context	5.111	12,06 %
<i>Film</i>	Environmental_context	2.500	5,90 %
<i>Humour</i>	Environmental_context	2.242	5,29 %
<i>Childhood</i>	Social_context	2.007	4,74 %
<i>Emotions</i>	Social_context	1.796	4,24 %
<i>Love</i>	Social_context	1.486	3,51 %
<i>Marriage</i>	Social_context	1.437	3,39 %
<i>Television</i>	Environmental_context	1.377	3,25 %
<i>Sports_entertainment</i>	Environmental_context	1.279	3,02 %
<i>Happiness</i>	Social_context	1.167	2,75 %
<i>Theatre</i>	Environmental_context	1.142	2,69 %
<i>Friendship</i>	Social_context	1.044	2,46 %
<i>Spouses</i>	Social_context	927	2,19 %
<i>Parts_of_a_day</i>	Time_context	898	2,12 %
<i>Barracks</i>	Location_context	769	1,81 %
<i>Total (top 15)</i>		25.182	59,42 %

**Tabla 5.11** Categorías contextuales más anotadas en las reviews de películas.

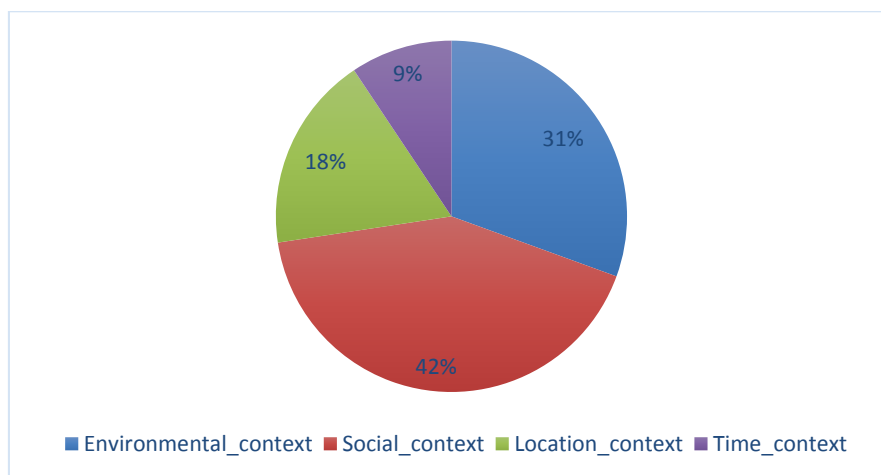
Las 15 categorías contextuales más anotadas representan más de la mitad del total de valores contextuales anotados, en concreto el 59,42% de todos ellos. En primer lugar, destaca la categoría ‘*Family*’, donde algunos de los valores contextuales anotados han sido palabras como ‘*daughter*’, ‘*son*’, ‘*family*’ o ‘*home*’, pudiendo identificar a priori con quién ha visto el usuario la película, algo que también permiten las anotaciones de la categoría ‘*Childhood*’, ‘*Friendship*’ o ‘*Spouses*’. También se puede identificar el lugar donde los usuarios vieron las películas, gracias a la anotación de categorías como ‘*Television*’ o ‘*Theather*’, que anotan palabras como ‘*TV*’, ‘*television*’, ‘*video*’ o ‘*theather*’ respectivamente; o la categoría ‘*Parts\_of\_a\_day*’, que etiquetando palabras como ‘*night*’, ‘*evening*’ o ‘*afternoon*’ permite identificar el momento del tiempo en el que el usuario ha visto dicha película.

Sin embargo, no todas las anotaciones realizadas son correctas; se puede ver como hay un gran número de anotaciones pertenecientes a la categoría contextual ‘*Humour*’ o ‘*Love*’, que en este caso no serían valores contextuales propiamente dichos, sino que seguramente el usuario esté realizando una descripción de la película; o anotaciones incorrectas como la categoría ‘*Film*’ que ha anotado la palabra ‘*movie*’ o ‘*film*’ como si fueran contextos, cuando lo más probable es que el usuario esté referenciando a la propia película con frases como ‘*In this movie...*’; o las categorías ‘*Marriage*’ y ‘*Sports\_entertainment*’.

Con todo ello, las dimensiones contextuales que más se repiten dentro de estos primeros valores contextuales son el contexto social y el entorno. Si se realiza el cálculo sobre el total de anotaciones, se obtiene que efectivamente las dimensiones contextuales más anotadas para el dominio de películas coinciden con lo que habíamos observado dentro de estas primeras categorías. En la siguiente tabla se puede observar la distribución del total de anotaciones contextuales sobre cada una de las cuatro dimensiones contextuales:

Dimensión contextual	Número de anotaciones	Porcentaje
<i>Environmental_context</i>	12.943	30,56 %
<i>Social_context</i>	17.810	42,05 %
<i>Location_context</i>	7.637	18,03 %
<i>Time_context</i>	3.966	9,36 %
<i>Total</i>	42.356	100 %

**Tabla 5.12** Anotaciones del dominio de películas según su dimensión contextual.



**Figura 5.7** Anotaciones para las reviews de películas según su dimensión contextual.

Esta distribución tiene cierto sentido, ya que al tratarse de reviews de películas, es normal que los usuarios detallen con quien vieron esa película, indicando por ejemplo si la vieron con sus amigos, con su pareja o con sus hijos; también es normal hacer alguna referencia al lugar donde vieron la película, indicando por ejemplo si la vieron en el cine o en TV, o indicando también el momento del tiempo en el que la vieron, diciendo si la vieron de noche o de día.

### 5.1.2 Evaluaciones textuales de música

A continuación se muestra el detalle para las evaluaciones textuales pertenecientes a música. En la siguiente tabla se muestra el número total de reviews, usuarios e ítems que fueron anotados y para aquellos de los cuales no fue posible realizar ninguna anotación.

	Total	Con contexto	Sin anotaciones	% anotación
<i>Número de reviews</i>	100.000	31.080	68.920	31,08 %
<i>Número de usuarios</i>	74.131	25.347	48.784	34.19 %
<i>Número de ítems</i>	6017	3.979	2.038	66,13 %

**Tabla 5.13** Reviews, usuarios e ítems para el dominio de música.

Para el dominio de música se ha conseguido anotar de contexto la mayor cantidad de reviews de entre los tres dominios, llegando a anotar algo más del 31% de las mismas. No obstante, el porcentaje de ítems anotados disminuye drásticamente si se compara con los otros dos dominios, alcanzando un 66% de ítems anotados con algún contexto frente a los altos porcentajes de los otros dominios. Sin embargo, aunque el porcentaje sea menor, el número de ítems que se encuentran dentro de este dominio es muy superior al de los otros.

Al igual que ocurría en el caso anterior, gran parte de los emparejamientos del dominio de música vienen representados por coincidencias exactas, o coincidencias con instancias o sinónimos de alguna instancia de DBpedia; representando entre estos tres niveles de matching un 78,4 % del total de anotaciones realizadas. En la siguientes tabla y figura se detallan los distintos niveles de matching empleados.

Nivel de matching	Número	Porcentaje
<i>Coincidencia exacta</i>	14.635	33,14 %
<i>Sinónimo</i>	4.304	9,75 %
<i>Instancia</i>	9.385	21,25 %
<i>Sinónimo de instancia</i>	10.604	24,01 %
<i>Redirección</i>	3.601	8,15 %
<i>Redirección desde sinónimo</i>	1.635	3,70 %
<i>Total anotaciones</i>	44.164	100 %

**Tabla 5.14** Niveles de matching empleados en las reviews de música.



**Figura 5.8** Niveles de matching empleados en las reviews de música.

Para este dominio, en total se han realizado 44.164 anotaciones contextuales, empleando un total de 305 categorías contextuales del total de categorías disponibles en la taxonomía. A continuación, se incluyen los 15 contextos anotados con mayor frecuencia para este dominio, indicando la dimensión contextual (rama de nivel 1 de la taxonomía) a la que pertenecen y el porcentaje que representan dichas anotaciones frente al número total de anotaciones realizadas para este dominio.

<b>Categoría contextual</b>	<b>Dimensión contextual</b>	<b>Número anotaciones</b>	<b>Porcentaje respecto al total</b>
<i>Music</i>	Environmental_context	5.988	13,56 %
<i>Family</i>	Social_context	2.869	6,50 %
<i>Love</i>	Social_context	2.591	5,87 %
<i>Straits</i>	Location_context	1.938	4,39 %
<i>Emotions</i>	Social_context	1.912	4,33 %
<i>Film</i>	Environmental_context	1.591	3,60 %
<i>Sports_entertainment</i>	Environmental_context	1.509	3,42 %
<i>Childhood</i>	Social_context	1.347	3,05 %
<i>Happiness</i>	Social_context	1.201	2,72 %
<i>Day</i>	Time_context	1.131	2,56 %
<i>Parts_of_a_day</i>	Time_context	1.125	2,55 %
<i>Marriage</i>	Social_context	1.043	2,36 %
<i>Earth</i>	Location_context	1.040	2,35 %
<i>Friendship</i>	Social_context	975	2,21 %
<i>Days</i>	Time_context	731	1,66 %
<i>Total (top 15)</i>		26.991	61,13 %

**Tabla 5.15** Categorías contextuales más anotados en las reviews de música.

Estas 15 categorías contextuales anotadas más frecuentemente dentro del dominio de música representan un 61,13 % del total de anotaciones de contexto realizadas en las reviews de dicho dominio, perteneciendo la mayor parte de ellas a la dimensión contextual de ‘*Social\_context*’ o bien de ‘*Environmental\_context*’. A nivel particular, se puede observar como algunos de los principales contextos anotados no son correctos, como ocurre por ejemplo con el contexto más anotado ‘*Music*’, que si bien la anotación en sí misma es correcta (ya que etiqueta la palabra ‘*music*’), dentro de la review no representa un contexto valido para estos ítems; o las anotaciones pertenecientes a la categoría ‘*Straits*’ o ‘*Sports\_entertainment*’ que están emparejando palabras como ‘*sound*’ o ‘*stars*’ respectivamente dentro de estas categorías, cuando en realidad dichas palabras ni están bien anotadas, ni representan anotaciones contextuales.

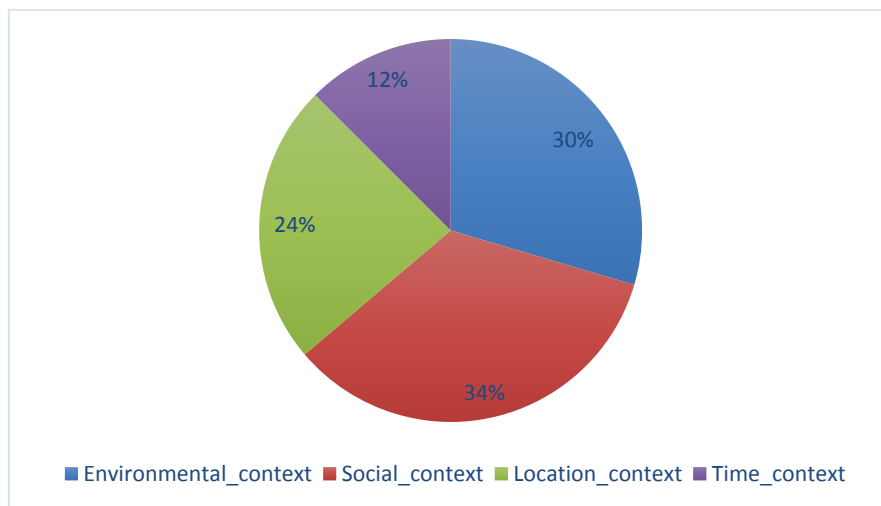
Por otro lado, se ha chequeado que las categorías de contexto obtenidas tienen sentido en el dominio de la música. Anotaciones que se realizan de forma correcta para las subcategorías de ‘*Emotions*’, como ‘*Happiness*’, han servido para identificar las emociones producidas en los usuarios por parte de los ítems, o ‘*Melancholia*’ que, aunque no se encuentra entre las más anotadas, representa un alto número de aciertos en este sentido. Otras categorías como ‘*Family*’, ‘*Childhood*’ o ‘*Marriage*’ también han servido para identificar con quien ha disfrutado el usuario de la música; o categorías

como ‘Day’ o ‘Parts\_of\_a\_day’ permiten identificar el momento del día más adecuado desde el punto de vista del usuario, para disfrutar de estos ítems.

En la siguiente tabla se muestra el total de anotaciones contextuales realizadas sobre las reviews de música, agrupándolas según la dimensión contextual a la que pertenecen:

Dimensión contextual	Número de anotaciones	Porcentaje
<i>Environmental_context</i>	13.058	29,59 %
<i>Social_context</i>	15.094	34,20 %
<i>Location_context</i>	10.468	23,72 %
<i>Time_context</i>	5.514	12,49 %
<i>Total</i>	42.356	100 %

**Tabla 5.16** Anotaciones del dominio de música según su dimensión contextual.



**Figura 5.9** Anotaciones para las reviews de música según su dimensión contextual.

Como se puede observar, las dimensiones contextuales se encuentran algo más repartidas que para el dominio de películas, no obstante, la dimensión social sigue siendo la más referenciada. Dentro de esta dimensión se encuentran las categorías que indican tanto con quien ha escuchado la música, como las distintas categorías que representan emociones que le han producido. Cabe destacar también la cantidad de anotaciones pertenecientes a la categoría de localización ‘*Location\_context*’, ya que representan donde los usuarios han escuchado los ítems de música.

### 5.1.3 Evaluaciones textuales de libros

En esta sección se detalla todo lo referente a las reviews pertenecientes al dominio de libros. En la siguiente tabla se muestra el número total de reviews, usuarios e ítems que han sido anotados con algún contexto, frente al total de aquellos para los cuales no ha sido posible realizar ninguna anotación.

	<b>Total</b>	<b>Con contexto</b>	<b>Sin anotaciones</b>	<b>% anotación</b>
<i>Número de reviews</i>	100.000	28.479	71.521	28,48 %
<i>Número de usuarios</i>	87.899	26.416	61.483	30,05 %
<i>Número de ítems</i>	377	325	52	86,21 %

**Tabla 5.17** Reviews, usuarios e ítems para el dominio de libros.

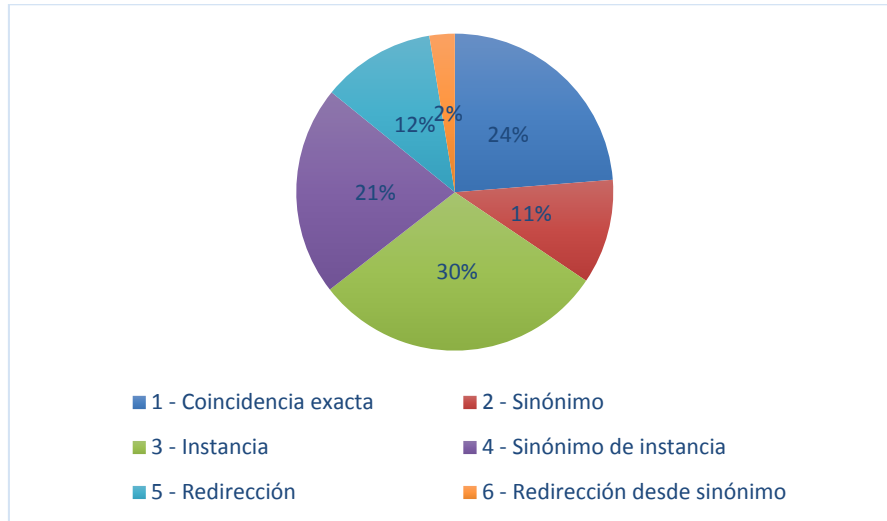
Para el dominio de libros, la cantidad de reviews anotadas es la menor de entre los tres dominios, habiendo podido anotar de contexto algo más del 28 % del total de reviews; no obstante, sigue estando cerca del total de reviews anotadas en los otros dominios. En cuanto al número de ítems anotados con contexto, se ha conseguido anotar un 86 % de los mismos, que, aunque no llega a igualarse con los ítems anotados en el dominio de películas (un 97 %), también representan una gran cantidad de los mismos.

Como ocurría con los dominios de películas y música, para reviews sobre libros la mayor parte de los emparejamientos de contexto vienen representados bien por coincidencias exactas, o bien por coincidencias con instancias obtenidas de DBpedia o sinónimos de dichas instancias. En concreto, estos tres tipos representan el 75,15 % del total de anotaciones realizadas. Así, para este dominio, el número de anotaciones de contexto realizadas a través de instancias obtenidas de DBpedia es el mayor de todos los dominios. En la siguientes tabla y figura se detallan los distintos tipos de emparejamiento empleados dentro de las reviews del dominio de libros.

<b>Nivel de matching</b>	<b>Número</b>	<b>Porcentaje</b>
<i>Coincidencia exacta</i>	9.383	23,75 %
<i>Sinónimo</i>	4.226	10,70 %
<i>Instancia</i>	11.859	30,02 %
<i>Sinónimo de instancia</i>	8.448	21,38 %
<i>Redirección</i>	4.577	11,58 %
<i>Redirección desde sinónimo</i>	1.017	2,57 %
<i>Total anotaciones</i>	39.510	100 %

**Tabla 5.18** Niveles de matching empleados en las reviews de libros.





**Figura 5.10** Niveles de matching empleados en las reviews de libros.

En total se han realizado 39.510 anotaciones contextuales, empleando 309 categorías contextuales disponibles en la taxonomía genérica de contexto, lo que representa la mayor cantidad de categorías contextuales anotadas de entre los tres dominios empleados.

En la tabla de abajo se muestran las 15 categorías contextuales que fueron anotadas con mayor frecuencia, indicando como en los casos anteriores, la dimensión contextual a la que pertenecen y el porcentaje de anotaciones que representa cada categoría de forma individual frente al total de anotaciones realizadas dentro de las reviews de este dominio.

Estas 15 categorías contextuales representan el 58,30 % del total de anotaciones de contexto realizadas dentro de las reviews de libros, y como se puede observar con claridad, la mayor parte de las mismas se encuadran dentro de la dimensión contextual '*Social\_context*' indicando principalmente el tipo de público al que va dirigido el libro, o para el que fue comprado. En concreto, se pueden ver ejemplos de categorías como '*Family*', donde fueron emparejadas palabras como '*daughter*', '*son*', '*father*' y '*mother*', todas ellas haciendo referencia al público objetivo del libro. Esto mismo ocurre con las categorías '*Childhood*' o '*Friendship*', que etiquetan palabras como '*girl*', '*boy*', '*child*', '*childhood*' o '*friends*' que de nuevo indican el público objetivo del libro. Otra categoría interesante es '*Schools*' que permite identificar el lugar donde fueron leídos los libros, en este caso en la escuela, o '*United\_States*' donde algunos de los emparejamientos hacen referencia a que los libros fueron leídos en los Estados Unidos, identificando palabras como '*America*' o '*US*'. De la misma forma que para los dominios anteriores, la categoría '*Parts\_of\_a\_day*' permite identificar el momento del día adecuado para los libros a los que hace referencia, emparejando (entre otras) palabras como '*morning*', '*afternoon*' o '*night*'. Por otro lado, se puede observar como algunos emparejamientos son erróneos a nivel de contexto dentro de las reviews. Esto

ocurre por ejemplo con la categoría ‘*Earth*’, donde se han anotado palabras como ‘*world*’, obtenida a través de un sinónimo de WordNet de la categoría ‘*Earth*’, como si fuera una anotación contextual. En este caso, la anotación es correcta, pero no representa ningún tipo de información contextual relevante.

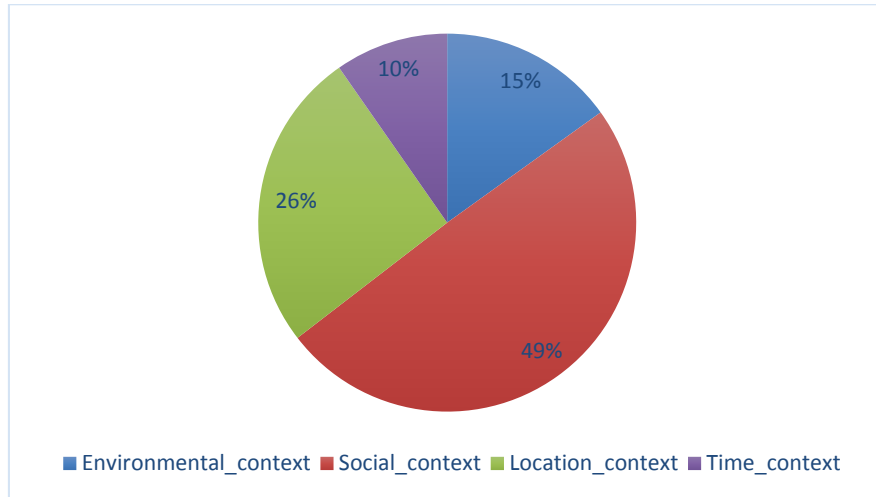
<b>Categoría contextual</b>	<b>Dimensión contextual</b>	<b>Número anotaciones</b>	<b>Porcentaje respecto al total</b>
<i>Family</i>	Social_context	4.412	11,17 %
<i>Emotions</i>	Social_context	2.848	7,21 %
<i>Marriage</i>	Social_context	2.478	6,27 %
<i>Childhood</i>	Social_context	1.783	4,51 %
<i>Earth</i>	Location_context	1.705	4,32 %
<i>Love</i>	Social_context	1.450	3,67 %
<i>Friendship</i>	Social_context	1.295	3,28 %
<i>Sports_entertainment</i>	Environmental_context	1.099	2,78 %
<i>Parts_of_a_day</i>	Time_context	997	2,52 %
<i>Happiness</i>	Social_context	928	2,35 %
<i>Schools</i>	Location_context	890	2,25 %
<i>United_States</i>	Location_context	846	2,14 %
<i>Humour</i>	Environmental_context	839	2,12 %
<i>Barracks</i>	Location_context	784	1,98 %
<i>Day</i>	Time_context	685	1,73 %
<i>Total (top 15)</i>		23.039	58,30 %

**Tabla 5.19** Categorías contextuales más anotados en las reviews de libros.

Como se ha mencionado, dentro de estas principales 15 categorías contextuales, la dimensión contextual más repetida es ‘*Social\_context*’, lo cual se repite si empleamos el conjunto total de anotaciones. En la siguiente tabla se muestra la distribución entre las cuatro dimensiones contextuales del total de anotaciones realizadas sobre las reviews de libros.

<b>Dimensión contextual</b>	<b>Número de anotaciones</b>	<b>Porcentaje</b>
<i>Environmental_context</i>	5.951	15,09 %
<i>Social_context</i>	19.512	49,47 %
<i>Location_context</i>	10.155	25,74 %
<i>Time_context</i>	3.826	9,70 %
<i>Total</i>	42.356	100 %

**Tabla 5.20** Anotaciones del dominio de libros según su dimensión contextual.



**Figura 5.11** Anotaciones para las reviews de libros según su dimensión contextual.

Como se puede observar, hay un claro predominio de la dimensión ‘*Social\_context*’, llegando a ocupar casi la mitad de las anotaciones realizadas dentro de las reviews de libros. Esto ocurre así porque es lógico que al realizar la review del libro, los usuarios incluyan información de a quién va dirigido el libro, como por ejemplo si lo compraron para regalar a un determinado pariente, si lo compraron para sus hijos, etc. de ahí que también otra parte de las anotaciones pertenezcan a la dimensión ‘*Location\_context*’, haciendo referencia donde fueron leídos esos libros.

## 5.2 Evaluación de las anotaciones de contexto

Una vez realizada la anotación contextual de las reviews de usuario y antes de proceder a aplicar los algoritmos de recomendación, se ha llevado a cabo una evaluación manual de dichas anotaciones para asegurar cierta calidad en el proceso de anotado, y de esta forma poder identificar posibles fallos además de posibles mejoras antes de continuar con la siguiente etapa.

Para ello, se ha implementado una herramienta de evaluación de anotaciones contextuales (cuyo diseño de base de datos puede encontrarse en el Anexo A de este documento), que permite a los usuarios cargar y revisar las distintas reviews anotadas contextualmente para su evaluación.

En la siguiente figura se muestra la aplicación de validación para una determinada review, donde además se ha anotado manualmente un contexto adicional identificado en la review:

**Current evaluation:**

Evaluation ID: 29

Item name: 0060092742

Summary: Great book - for any age

Review: I greatly enjoyed reading this book. The writing is fabulous and engaging. Anyone can identify with the subject matter. I recommended it to my **Mother**, and to my teenaged nieces.

**Automatic context annotations:**

ID	Keyword (in review)	Context (in taxonomy)	Match level	Match similarity	Is context?	Is correct?
5060	Mother	Family	Instance	1.0	Yes	Yes

**Context taxonomy:**

Instances and synonyms of select taxonomy:

Filter word: niece

Word	Type
Nephew and niece	3 - Instance

**Manual context annotations:**

ID	Keyword (in review)	Context (in taxonomy)
8	nieces	Family

Figura 5.12 Herramienta de evaluación de anotaciones de contexto.

Como se puede observar, la herramienta cuenta con cuatro zonas diferenciadas. En la esquina superior izquierda, la zona titulada como *'Current evaluation'*, se muestra la propia review del usuario que se está evaluando, junto con los metadatos del ítem al que pertenece. Dentro del texto de la review, las palabras que han sido identificadas como contexto se muestran resaltadas para que sea más fácil identificarlas.

En la esquina superior derecha, la zona titulada como *'Automatic context annotations'*, muestra una tabla con las distintas anotaciones de contexto identificadas dentro del texto. En esta tabla se muestra la palabra (que a su vez se muestra resaltada dentro del texto de la review), la categoría de contexto a la que pertenece dentro de la taxonomía, el tipo de anotación (categoría, instancia, sinónimo, etc.) y el *score* de similitud correspondiente a la anotación de contexto. A sí mismo, el usuario puede validar cada una de estas anotaciones indicando en primer lugar si la palabra anotada es contexto o no, y en caso de tratarse de contexto, indicar si la anotación es correcta, incorrecta o sería mejor si se etiquetase con una categoría hija o padre de la categoría asignada dentro de la taxonomía de contexto.

En la parte inferior, en la esquina inferior derecha, la zona titulada como *'Manual context annotations'*, el usuario puede incluir manualmente anotaciones de contexto que haya sido capaz de identificar dentro del texto de la review y que o bien no haya sido anotada de forma automática, o bien la anotación sea incorrecta y sea necesario especificar la categoría contextual correcta a la que pertenece. Por último, en la esquina inferior izquierda, la zona denominada *'Context taxonomy'* muestra la taxonomía genérica de contexto, junto con sus instancias y sinónimos, para permitir una búsqueda de manera interactiva.

Debido a la gran cantidad de reviews disponibles para los tres dominios y de anotaciones contextuales que se disponen, se han seleccionado un subconjunto de las

mismas para llevar a cabo la evaluación manual empleando la herramienta de evaluación explicada anteriormente. En total, se han seleccionado 10 ítems de cada uno de los dominios (30 ítems en total), y se han seleccionado las reviews correspondientes a esos ítems (descartando aquellas reviews que no tuvieran ninguna anotación) lo que hace un total de 148 reviews de películas, 57 reviews de música y 86 reviews de libros.

Después, para cada una de esas reviews, se han seleccionado todas sus anotaciones contextuales, lo que representa un total de 272 anotaciones dentro de esas 148 reviews de películas, 139 anotaciones dentro de las 57 reviews de música y 151 anotaciones dentro de las 86 reviews de libros, para evaluar en todas ellas si por un lado la anotación es correcta, es decir, que el emparejamiento entre la categoría contextual y la palabra de la review sea correcto y tiene sentido; y por otro lado, si la anotación representa contexto, es decir, si se está haciendo referencia a algún aspecto contextual a través de esa palabra de la review.

Al mismo tiempo que se realiza la evaluación de dichas anotaciones y reviews, en algunas de las reviews se han descubierto nuevas anotaciones contextuales que no habían sido encontradas de forma automática, por lo que han sido anotadas de manera manual. En concreto se han encontrado 11 anotaciones contextuales dentro de las reviews de los libros, 15 dentro de las reviews de películas y 15 dentro de las reviews de música.

En la siguiente tabla se muestra la información referente a este proceso de evaluación manual, junto con los porcentajes de anotaciones correctas y de anotaciones contextuales correctas, teniendo en cuenta todas las reviews y anotaciones evaluadas, sin ningún tipo de agrupación.

	<b>Películas</b>	<b>Música</b>	<b>Libros</b>
<i>Items seleccionados</i>	10	10	10
<i>Reviews evaluadas</i>	148	57	86
<i>Anotaciones evaluadas</i>	272	139	151
<i>Anotaciones correctas (Sí/No)</i>	199/73	94/44	99/52
<i>% de anotaciones correctas</i>	73,16 %	68,35 %	65,56 %
<i>Anotaciones contextos correctas (Sí/No)</i>	84/188	49/90	31/120
<i>% de contextos correctos</i>	30,88 %	35,25 %	20,53 %
<i>Número anotaciones manuales</i>	15	15	11

**Tabla 5.21** Tabla resumen tras la evaluación manual de anotaciones.

Como se puede observar, el porcentaje de anotaciones correctas es relativamente elevado, un 69,02 % de media, mientras que, por otro lado, el porcentaje de anotaciones de contexto correctas es bajo, representando de media solo un 28,89 % de todas las anotaciones.

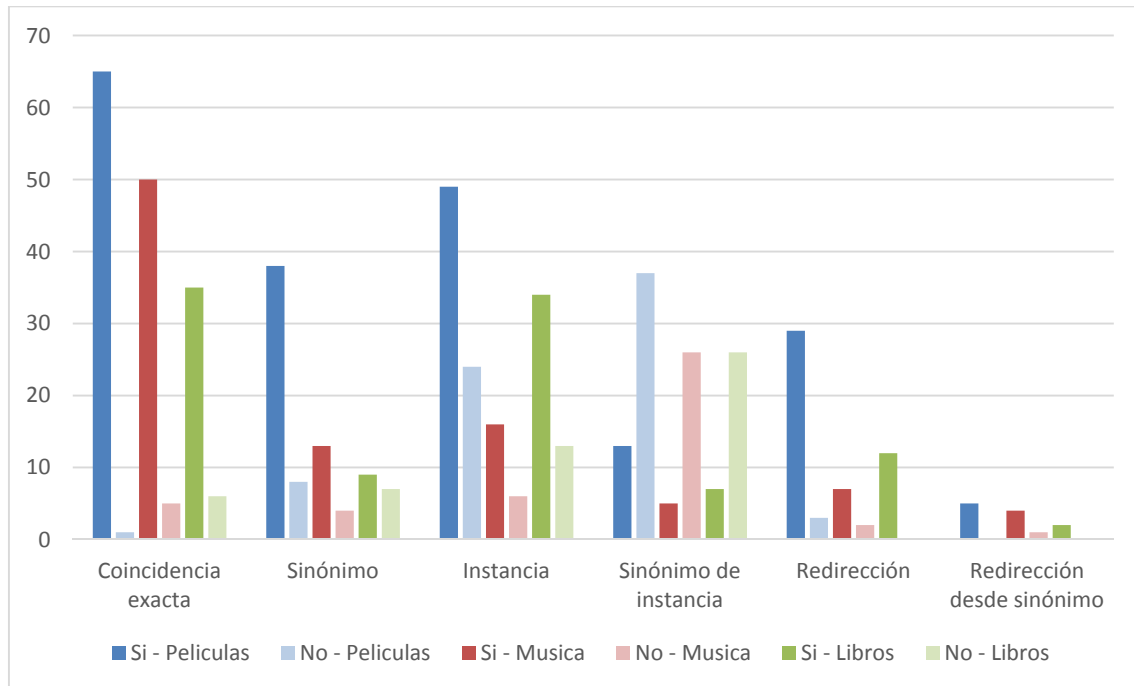
A continuación se llevan a cabo una serie de agrupaciones en torno a las anotaciones evaluadas manualmente para ver el porcentaje de anotaciones correctas y de anotaciones con contexto correctas dependiendo del nivel de matching llevado a cabo o de la dimensión contextual a la que pertenecen las anotaciones, para de esta forma desglosar los resultados obtenidos e intentar encontrar los puntos donde el proceso de anotación sea más efectivo.

En la siguiente tabla se muestran los resultados agrupados según si las anotaciones han sido correctas o no y teniendo en cuenta el nivel de matching al que pertenecen cada una de las anotaciones realizadas en las reviews.

Anotaciones correctas según su nivel de matching	Películas			Música			Libros		
	Sí	No	Porcentaje correctas	Sí	No	Porcentaje correctas	Sí	No	Porcentaje correctas
<i>Coincidencia exacta</i>	65	1	98,48 %	50	5	90,91 %	35	6	85,37 %
<i>Sinónimo</i>	38	8	82,61 %	13	4	76,47 %	9	7	56,25 %
<i>Instancia</i>	49	24	67,12 %	16	6	72,73 %	34	13	72,34 %
<i>Sinónimo de instancia</i>	13	37	26 %	5	26	16,13 %	7	26	21,21 %
<i>Redirección</i>	29	3	90,63 %	7	2	77,78 %	12	0	100 %
<i>Redirección desde sinónimo</i>	5	0	100 %	4	1	80 %	2	0	100 %
<i>Total anotaciones</i>	199	73	73,16 %	95	44	68,35 %	99	52	65,56 %

**Tabla 5.22** Tabla comparativa anotaciones correctas agrupadas según nivel de matching.

En la siguiente gráfica se muestran los datos de las tres tablas anteriores de manera conjunta, para obtener una visión global del porcentaje de anotaciones correctas agrupadas por el nivel de matching empleado y según el dominio de los datos.



**Figura 5.13** Anotaciones correctas (Sí/No) por nivel de matching y dominio.

Al llevar a cabo esta agrupación se puede distinguir con mayor claridad que niveles de matching son los que resultan más efectivos, frente a aquellos que en su mayoría producen anotaciones incorrectas. En concreto, la coincidencia exacta, la instancia de DBpedia y la redirección a una instancia de DBpedia, son en líneas generales los tres niveles de matching que producen un mayor número de anotaciones correctas para los tres datasets. De hecho, utilizando únicamente estos tres niveles de matching, el porcentaje de anotaciones correctas se incrementa con notoriedad, como puede observarse en la siguiente tabla, alcanzando una media de anotaciones correctas del 84,17 %, frente al 69,02 % de anotaciones correctas que se tiene si se emplean todos los tipos de emparejamiento.

	Películas	Música	Libros
<i>Anotaciones evaluadas</i>	272	139	151
<i>Anotaciones correctas (Sí/No)</i>	199/73	94/44	99/52
<i>% de anotaciones correctas</i>	73,16 %	68,35 %	65,56 %
<i>Anotaciones evaluadas solo si coincidencia exacta, instancia o redirección a instancia</i>	171	86	100
<i>Anotaciones correctas (Sí/No)</i>	143/28	73/13	81/19
<i>% de anotaciones correctas</i>	86,63 %	84,89 %	81 %

**Tabla 5.23** Tabla comparativa anotaciones correctas empleando sólo emparejamientos por coincidencia exacta, instancias de DBpedia y redirecciones a instancias.

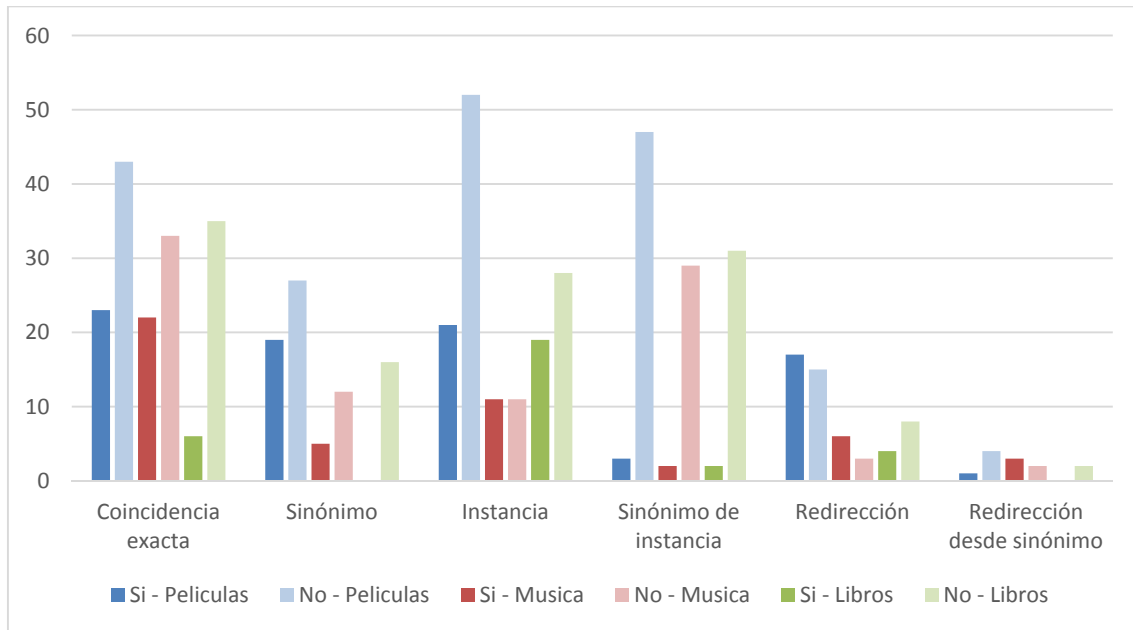
También se puede observar como los sinónimos de WordNet, para los dominios de películas y música, producen bastantes anotaciones correctas, mientras que para el dominio de libros, los resultados están bastante equilibrados entre anotaciones por sinónimo correctas e incorrectas. Por último, se puede ver como las anotaciones llevadas a cabo por sinónimos de instancias apenas producen anotaciones correctas, ya que para el mejor de los casos (películas) apenas suponen un 26 % de anotaciones correctas, lo cual representa un porcentaje muy bajo comparado con los otros niveles de matching.

A continuación se lleva a cabo una agrupación similar por nivel de matching, pero teniendo en cuenta si las anotaciones contextuales son correctas o no, es decir, si la anotación en si misma representa contexto. De la misma forma, se presentarán los resultados en tres tablas, una para cada dominio de reviews, y posteriormente una gráfica con estos datos agrupados.

Anotaciones contexto correctas según el nivel de matching	Películas			Música			Libros		
	Sí	No	Porcentaje correctas	Sí	No	Porcentaje correctas	Sí	No	Porcentaje correctas
<i>Coincidencia exacta</i>	23	43	34,85 %	22	33	40 %	6	35	14,63 %
<i>Sinónimo</i>	19	27	41,3 %	5	12	29,41 %	0	16	0 %
<i>Instancia</i>	21	52	28,77 %	11	11	50 %	19	28	40,43 %
<i>Sinónimo de instancia</i>	3	47	6 %	2	29	6,45 %	2	31	6,06 %
<i>Redirección</i>	17	15	53,13 %	6	3	66,67 %	4	8	33,33 %
<i>Redir. desde sinónimo</i>	1	4	20 %	3	2	60 %	0	2	0 %
<i>Total anotaciones</i>	84	188	30,88 %	49	90	35,25 %	31	120	20,53 %

**Tabla 5.24** Tabla comparativa anotaciones contexto correctas agrupadas según nivel de matching.





**Figura 5.14** Anotaciones contexto correctas (Sí/No) por nivel de matching y dominio.

En cuanto a la anotación correcta de contexto, se obtiene que los porcentajes son relativamente bajos a nivel general, no obstante, al igual que ocurría con las anotaciones correctas, puede observarse como determinados niveles de matching producen mejores resultados a la hora de realizar anotaciones de contexto. Se puede observar como los emparejamientos por coincidencia exacta producen buenos resultados tanto para películas como para música, pero para libros los resultados son bastante peores con este tipo de emparejamiento.

Los sinónimos de WordNet producen malos resultados tanto para música como para libros, en cambio para el dominio de películas, el 41% de las anotaciones de contexto por sinónimo son correctas, lo cual es un porcentaje bastante alto. En cambio, las instancias de DBpedia producen resultados más bajos en el dominio de películas, pero bastante buenos dentro de la música y los libros.

Al igual que ocurría con las anotaciones correctas, las anotaciones de contexto llevadas a cabo a través de sinónimos de instancia producen resultados que están muy lejos de ser aceptables. En el mejor de los casos (música), únicamente el 6,45% de las anotaciones contextuales con este tipo de emparejamiento han sido correctas.

Por último, se puede ver como las anotaciones de contexto llevadas a cabo a través de redirecciones a instancias de DBpedia producen resultados bastante buenos, teniendo en cuenta el hecho de que este tipo de anotación se produce en un número más reducido de casos.

Para poder llevar a cabo una comparativa similar a la realizada anteriormente con el porcentaje de anotaciones correctas, se han seleccionado aquellas anotaciones contextuales que habían sido realizadas por los mismos tipos de emparejamiento

empleados que producían los mejores resultados, en concreto, aquellos realizados por coincidencia exacta de la categoría, por instancias de DBpedia o por redirección a una instancia. En la siguiente tabla se compara el porcentaje de anotaciones contextuales correctas total, frente a empleando sólo aquellas realizadas por estos tres tipos de emparejamiento.

	<b>Películas</b>	<b>Música</b>	<b>Libros</b>
<i>Anotaciones evaluadas</i>	272	139	151
<i>Anotaciones contextos correctas (Sí/No)</i>	84/188	49/90	31/120
<i>% de contextos correctos</i>	30,88 %	35,25 %	20,53 %
<i>Anotaciones evaluadas solo si coincidencia exacta, instancia o redirección a instancia</i>	171	86	100
<i>Anotaciones contextos correctas (Sí/No)</i>	61/110	39/47	29/71
<i>% de contextos correctos</i>	35,67 %	45,34 %	29 %

**Tabla 5.25** Tabla comparativa anotaciones contextuales correctas empleando sólo emparejamientos por coincidencia exacta, instancias de DBpedia y redirecciones a instancias.

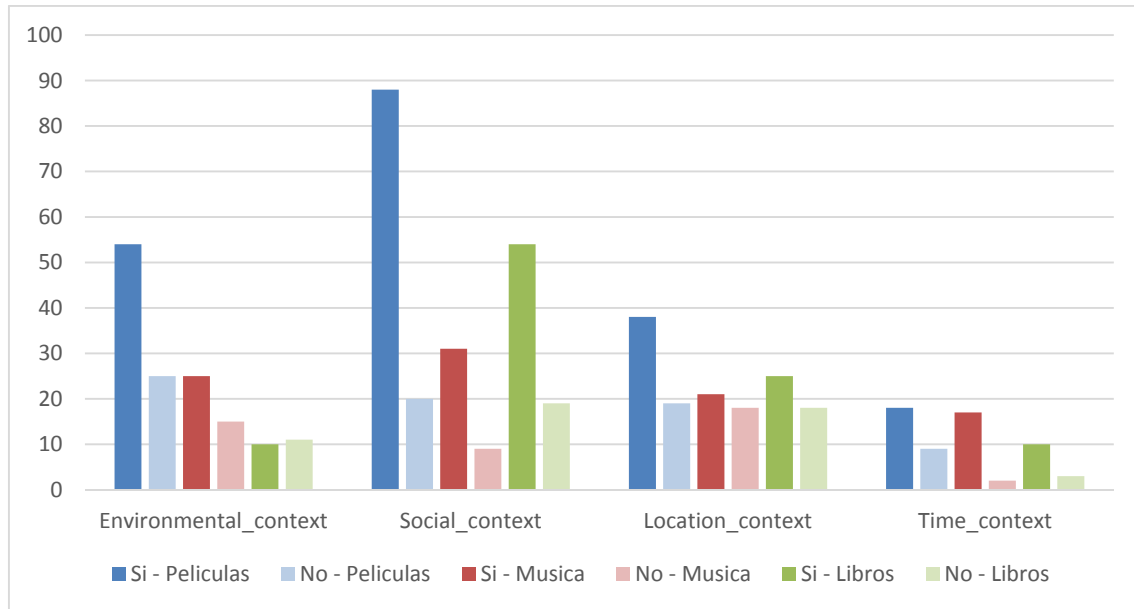
Empleando sólo las anotaciones de estos tres tipos de emparejamiento, el porcentaje de anotaciones contextuales correctas se incrementa, pasando del 28,89% de media que teníamos al emplear todas las anotaciones a un 36,67% de anotaciones contextuales correctas si se emplean solo estos tres niveles de matching, no obstante, este porcentaje todavía puede considerarse relativamente bajo, ya que significa que solo una de cada tres anotaciones contextuales es correcta.

A continuación se lleva a cabo una agrupación de las anotaciones según la dimensión contextual a la que pertenecen, indicando cuales de ellas son correctas y cuáles no. En la siguiente tabla se muestran estos resultados para los tres dominios de reviews disponibles, incluyendo la media de los tres porcentajes obtenidos.

<b>Anotaciones correctas (Sí/No) según su dimensión contextual</b>	<b>Películas</b>			<b>Música</b>			<b>Libros</b>			<b>Media</b>
	<b>Sí</b>	<b>No</b>	<b>Porcentaje correctas</b>	<b>Sí</b>	<b>No</b>	<b>Porcentaje correctas</b>	<b>Sí</b>	<b>No</b>	<b>Porcentaje correctas</b>	
<i>Environmental_context</i>	54	25	68,35 %	25	15	62,5 %	10	11	47,62 %	59,49%
<i>Social_context</i>	88	20	81,48 %	31	9	77,5 %	54	19	73,97 %	77,65%
<i>Location_context</i>	38	19	66,67 %	21	18	53,85 %	25	18	58,14 %	59,55%
<i>Time_context</i>	18	9	66,67 %	17	2	89,47 %	10	3	76,92 %	77,69%
<i>Total</i>	198	73	73,06 %	94	44	68,12 %	99	51	66 %	69,06%

**Tabla 5.26** Tabla comparativa anotaciones correctas agrupadas según la dimensión contextual.

El número total de anotaciones es inferior que para los casos anteriores, concretamente se diferencian en 1 anotación. Esto es debido a que dicha anotación fue anotada como ‘Contexto’ en general, por lo que no pertenece a ninguna de las dimensiones contextuales que estamos considerando. En la siguiente figura se muestra la información de la tabla anterior de manera visual.



**Figura 5.15** Anotaciones correctas (Sí/No) por tipo dimensión contextual y dominio reviews.

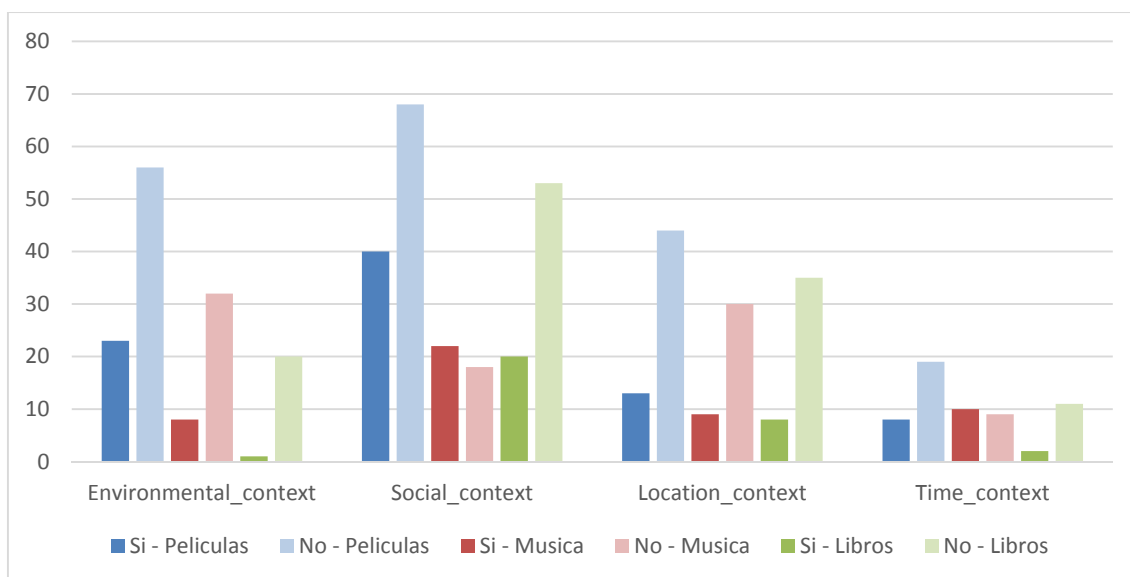
A nivel global, los mejores resultados en cuanto a anotación se obtienen para las dimensiones contextuales de ‘*Social\_context*’ y ‘*Time\_context*’, alcanzando de media del 77,65% y el 77,69% de anotaciones correctas, llegando a superar el 80% de anotaciones correctas para alguno de los dominios correspondientes, como es el caso de ‘*Social\_context*’ en películas y ‘*Time\_context*’ en las anotaciones de música.

Las otras dos dimensiones contextuales ‘*Environmental\_context*’ y ‘*Location\_context*’, obtienen resultados inferiores en cuanto a porcentaje de anotación. No obstante, para ambos casos se consigue casi el 60% de anotaciones correctas, lo cual supone un porcentaje de anotación correcta relativamente bueno. Sólo para el dominio de películas se han obtenido resultados significativamente superiores para estas dos dimensiones contextuales, 68,35% y 66,67% respectivamente.

Por último, se ha llevado a cabo la misma agrupación según la dimensión contextual de las distintas anotaciones, pero esta vez teniendo en cuenta cuales son correctas desde el punto de vista de anotaciones contextuales, y cuáles no. En la siguiente tabla se muestran los resultados obtenidos tras llevar a cabo esta agrupación, incluyendo el porcentaje de anotaciones contextuales correctas, y la media de dichos porcentajes para cada una de las dimensiones contextuales de la taxonomía. De la misma manera, se incluye una gráfica con la información de la tabla de una forma visual.

Anotaciones contexto correctas (Sí/No) según su dimensión contextual	Películas			Música			Libros			Media
	Sí	No	Porcentaje correctas	Sí	No	Porcentaje correctas	Sí	No	Porcentaje correctas	
<i>Environmental_context</i>	23	56	29,11 %	8	32	20 %	1	20	4,76 %	17,96%
<i>Social_context</i>	40	68	37,04 %	22	18	55 %	20	53	27,4 %	39,81%
<i>Location_context</i>	13	44	22,81 %	9	30	23,08 %	8	35	18,6 %	21,5%
<i>Time_context</i>	8	19	29,63 %	10	9	52,63 %	2	11	15,38 %	32,55%
<i>Total</i>	84	187	31 %	49	89	35,51 %	31	119	20,67 %	29,06%

**Tabla 5.27** Tabla comparativa anotaciones contexto correctas agrupadas según la dimensión contextual.



**Figura 5.16** Anotaciones contexto correctas (Sí/No) por tipo dimensión contextual y dominio.

De manera similar a como ocurría antes, los resultados más prometedores en lo que anotación contextual se refiere, se encuentra dentro de las categorías ‘*Social\_context*’ y ‘*Time\_context*’, obteniendo de media un 39,81% y 32,55% de anotaciones contextuales correctas respectivamente. Desde este punto de vista, cabe destacar como para el dominio de libros, las anotaciones contextuales correctas dentro de estas dos dimensiones representan un porcentaje prometedor, en concreto el 55% y el 52,63% respectivamente.

En cambio, las otras dos dimensiones contextuales ‘*Environmental\_context*’ y ‘*Location\_context*’, se encuentran en cotas de acierto muy bajas. Por ejemplo, para el caso del dominio de libros, solo se anotaron de forma correcta el 4,76 % de las anotaciones contextuales de la dimensión ‘*Environmental\_context*’.

### 5.3 Evaluación de las recomendaciones contextualizadas

Una vez realizada la evaluación de las anotaciones de contexto, se llevaron a cabo distintos experimentos de recomendación sobre dichas anotaciones, para comprobar si el contexto anotado a partir de las reviews de usuario, representa una mejora en el proceso de anotación frente a los métodos tradicionales que no tienen en cuenta los factores contextuales.

Como se ha observado en la fase de evaluación de las anotaciones, no todas las anotaciones realizadas son correctas, por esta razón es necesario establecer qué variables contextuales (y sus valores) serán tenidos en cuenta a la hora de realizar los experimentos de recomendación, ya que, si se utilizase el conjunto total de datos anotados, se estaría incluyendo demasiado ruido (categorías mal anotadas, errores en la anotación de palabras, etc.), y los experimentos de recomendación no darían resultados válidos.

Para ello, tras analizar el proceso de anotación y realizar la evaluación manual de algunas de las anotaciones, se han seleccionado un total de 5 categorías contextuales de la taxonomía general, para realizar los experimentos centrándose por separado en cada una de esas categorías y sus valores, como si se tratasen de conjuntos anotados de manera independiente. Por último, esas 5 categorías contextuales se han unido en un último conjunto de test para realizar los mismos experimentos, pero esta vez sobre un conjunto anotado que contuviera las cinco categorías. Las categorías seleccionadas son las siguientes:

- ***Interpersonal\_relationship***: esta categoría pertenece a la dimensión contextual '*Social\_context*', para la cual se habían obtenido resultados prometedores en acierto de anotación contextual. Para crear este subconjunto de datos, se han utilizado todas aquellas anotaciones contextuales que tuvieran alguna subcategoría de *Interpersonal\_relationship*, y dichas subcategorías se han utilizado como los posibles valores a considerar para la variable contextual, de esta forma, los posibles valores de esta categoría utilizados en la recomendación son *Familiy*, *Friendship*, *Marriage*, etc. Si la anotación contextual pertenece a otra categoría diferente, se tratará como si no tuviese ningún tipo de contexto asociado.
- ***Emotions***: de manera similar a la categoría anterior, esta también pertenece a la dimensión contextual '*Social\_context*', y también se han utilizado sus distintas subcategorías para construir el conjunto de datos. Por ello, los distintos valores que puede tomar esta variable contextual '*Emotions*' a la hora de aplicar las recomendaciones son las propias subcategorías como por ejemplo *Love*, *Sadness*, *Happiness*, *Suffering* etc.

- **Academic\_institutions:** a pesar de que esta categoría pertenece a la dimensión ‘*Location\_context*’ que en la evaluación manual se ha visto que tenía cotas de acierto muy bajas respecto a otras categorías, se ha decidido utilizar ya que, tras analizar sus subcategorías de manera particular, se puede pensar que están anotadas de forma correcta. Estas subcategorías son *Academic\_institutions*, *Schools* y *Universities\_and\_colleges*, y corresponderán los posibles valores de la variable contextual empleada en la recomendación.
- **Parts\_of\_a\_day:** esta categoría contextual pertenece a la dimensión ‘*Time\_context*’, que junto con ‘*Social\_context*’ también se habían obtenidos resultados prometedores durante la evaluación de las anotaciones. En este caso, para los posibles valores a emplear en la recomendación no se han empleado las subcategorías de ‘*Parts\_of\_a\_day*’ ya que no tiene, sino que se han empleado las distintas palabras anotadas dentro de la review del usuario como pertenecientes a esta categoría. Así, algunos de los posibles valores para este contexto empleados en la recomendación son *afternoon*, *dark*, *morning*, *night*, etc.
- **Working\_time:** esta categoría también pertenece a la dimensión ‘*Time\_context*’, y al igual que con ‘*Parts\_of\_a\_day*’, se han utilizado las palabras anotadas bajo esta categoría como los valores de la variable contextual a la hora de realizar la recomendación. En este caso, solo pueden existir dos valores posibles para esta variable y son *week* y *weekend*.

Como se ha comentado, además de estos 5 conjuntos de datos, se ha generado uno más con el contenido de las cinco variables contextuales, y este proceso se ha repetido para cada uno de los 3 dominios de datos (película, música y libros) que existían. Por tanto, en total se han generado 18 conjuntos de datos sobre los cuales se aplicarán los distintos experimentos de recomendación. En las siguientes tablas se pueden ver las características de estos conjuntos de datos para cada dominio, indicando para el total de reviews, usuarios e ítems, cuántos de ellos tienen anotaciones contextuales.

	Total datos	Anotaciones contextuales según conjunto de datos de películas					
		Interpersonal	Emotions	Academic	Parts of a day	Working time	Todos juntos
<i>Reviews</i>	100.000	10.439 (10,44 %)	5.435 (5,43 %)	503 (0,50 %)	874 (0,87 %)	288 (0,29 %)	15.983 (15,98 %)
<i>Usuarios</i>	77.030	9.720 (12,62 %)	4.994 (6,48 %)	491 (0,64 %)	854 (1,11 %)	288 (0,37 %)	14.340 (18,61 %)
<i>Ítems</i>	323	301 (93,19 %)	296 (91,64 %)	173 (53,56 %)	246 (76,16 %)	147 (45,51 %)	306 (94,73 %)

**Tabla 5.28** Características conjuntos de datos para el dominio de películas.

	Total datos	Anotaciones contextuales según conjunto de datos de música					
		Interpersonal	Emotions	Academic	Parts of a day	Working time	Todos juntos
<i>Reviews</i>	100.000	7.153 (7,15 %)	6.619 (6,62 %)	287 (0,29 %)	1.086 (1,09 %)	472 (0,47 %)	14.194 (14,19 %)
<i>Usuarios</i>	74.131	6.543 (8,82 %)	5.904 (7,96 %)	286 (0,39 %)	1.043 (1,41 %)	460 (0,62 %)	12.332 (16,64 %)
<i>Ítems</i>	6.017	1.639 (27,23 %)	1.814 (30,15 %)	210 (3,49 %)	543 (9,02 %)	217 (3,61 %)	2.540 (42,21 %)

**Tabla 5.29** Características conjuntos de datos para el dominio de música.

	Total datos	Anotaciones contextuales según conjunto de datos de libros					
		Interpersonal	Emotions	Academic	Parts of a day	Working time	Todos juntos
<i>Reviews</i>	100.000	10.406 (10,41 %)	6.590 (6,59 %)	1.065 (1,06 %)	956 (0,96 %)	272 (0,27 %)	17.543 (17,54 %)
<i>Usuarios</i>	87.899	10.073 (11,46 %)	6.390 (7,27 %)	1.055 (1,20 %)	950 (1,08 %)	272 (0,31 %)	16.701 (19,00 %)
<i>Ítems</i>	377	294 (77,99 %)	264 (70,03 %)	151 (40,05 %)	178 (47,21 %)	88 (23,34 %)	307 (81,43 %)

**Tabla 5.30** Características conjuntos de datos para el dominio de libros.

Como se puede observar en las tablas anteriores, el porcentaje de información contextual es bastante pequeño para los distintos conjuntos de datos, lo que hace más difícil generar buenas recomendaciones mediante métodos contextuales debido a la poca información contextual que poseen.

Estos experimentos se evaluarán desde el punto de vista de la predicción del rating de ítems, es decir, midiendo el error cometido (MAE y RMSE) a la hora de predecir los ratings. Se ha optado por este tipo de evaluación frente a una evaluación del ranking de ítems ya que son pocos los usuarios que han evaluado varios ítems, lo que hace difícil poder establecer qué ítems son relevantes (y en qué grado) para los distintos usuarios.

Todas las evaluaciones han sido realizadas mediante un proceso de validación cruzada de 5 iteraciones (*5-folds cross-validation*) habiendo mezclado previamente de forma aleatoria el conjunto de datos. Este proceso consiste en que una vez mezclados los datos de forma aleatoria, se dividen en 5 subconjuntos de igual longitud, de forma que se emplea uno de los subconjuntos como datos de test y los otros 4 restantes como datos de entrenamiento en el proceso de recomendación. Este proceso se repite durante 5 iteraciones, utilizando como subconjunto de test cada uno de los 5 subconjuntos generados. Finalmente, se realiza la media aritmética de los resultados obtenidos en cada iteración y con ello obtendríamos el resultado de la evaluación del algoritmo de recomendación evaluado sobre el correspondiente conjunto de datos. No obstante, este procedimiento no está libre de problemas, ya que es posible que sufra el problema de *Cold Start* o arranque en frío, si por ejemplo se escogen aleatoriamente todos los ratings

de un usuario dentro del conjunto de test y ninguno para entrenamiento. Esto puede producir que, a la hora de realizar la evaluación, el modelo no tenga información previa de ese usuario y las recomendaciones no sean del todo adecuadas.

Los métodos para evaluar las recomendaciones son los métodos que se detallaron en la Sección 4.3, y que se resumen a continuación en la siguiente tabla:

Métodos <i>baseline</i>	Métodos conscientes del contexto	
GlobalAverage	CAMF_C	ItemSplitting(IS) + BiasedMF
UserAverage	CAMF_CI	ItemSplitting(IS) + ItemKNN
ItemAverage	CAMF_CU	ItemSplitting(IS) + UserKNN
UserKNN	CAMF_CUCI	UISplitting(UIS) + BiasedMF
ItemKNN		UISplitting(UIS) + ItemKNN
BiasedMF		UISplitting(UIS) + UserKNN

**Tabla 5.31** Tabla con los métodos de recomendación evaluados.

### 5.3.1 Recomendación sobre películas

En la siguiente tabla se muestran los resultados obtenidos al aplicar los métodos de referencia o *baseline* sobre los conjuntos de datos anotados de películas. Como estos métodos no tienen en cuenta el valor contextual, los resultados serán los mismos para los 6 conjuntos de datos que se van a analizar, ya que la única diferencia de ellos radica en los valores contextuales añadidos, por tanto, estos resultados son válidos de forma general para cualquier conjunto de datos sobre el dominio de películas. Se han resaltado aquellos valores que dan menor MAE y menor RMSE para tomarlos como referencia para comparar frente a los métodos conscientes del contexto.

<i>Baseline</i>	GlobalAverage	UserAverage	ItemAverage	UserKNN	ItemKNN	BiasedMF
<i>MAE</i>	0.8929	0.8629	0.8346	0.8851	0.8566	<b>0.8242</b>
<i>RMSE</i>	1.1630	1.1853	<b>1.1184</b>	1.1590	1.1623	1.1250

**Tabla 5.32** Resultados en la predicción de rating sobre películas empleando métodos *baseline*.

En este caso, el menor error medio absoluto (MAE) viene proporcionado al utilizar el método de factorización de matrices (BiasedMF), mientras que el menor error cuadrático medio (RMSE) viene proporcionado al utilizar directamente la media de los ítems (ItemAverage) como método de recomendación.

En la siguiente tabla se muestran los resultados de realizar las evaluaciones de cada uno de los métodos conscientes del contexto sobre cada uno de los 6 conjuntos de datos generados con los distintos valores contextuales a analizar y la unión de todos ellos, como se ha explicado al inicio de la sección.

Se han resaltado aquellos resultados que permiten asegurar que la información contextual añadida mejora los resultados obtenidos en la recomendación. Esto se puede obtener de dos maneras, la **primera** de ellas es obtener menor MAE o RMSE que los



mejores casos de los valores de referencia o *baseline*, y la **segunda** de ellas es que los métodos que emplean *Item Splitting* o *UI Splitting*, den mejores resultados que sus métodos de referencia equivalentes, ya que esto también indica que la información contextual ha sido aprovechada en beneficio de la recomendación, en este aspecto también se han **resaltado** aquellos resultados que tras aplicar el *splitting* producen los mismos resultados que si no se aplicase la separación.

		CAMF C	CAMF CI	CAMF CU	CAMF CUCI	IS + BiasedMF	IS + ItemKNN	IS + UserKNN	UIS + BiasedMF	UIS + ItemKNN	UIS + UserKNN
MAE	Interpersonal	<b>0.8236</b>	0.8392	<b>0.7958</b>	0.8485	<b>0.8222</b>	0.8567	0.8853	<b>0.8217</b>	0.8567	0.8852
	Emotions	<b>0.8233</b>	0.8399	<b>0.7949</b>	0.8483	<b>0.8227</b>	<b>0.8565</b>	0.8851	<b>0.8217</b>	0.8567	0.8851
	Academic	<b>0.8233</b>	0.8432	<b>0.7936</b>	0.8470	<b>0.8207</b>	0.8566	0.8851	<b>0.8221</b>	0.8566	0.8851
	Parts of day	<b>0.8238</b>	0.8432	<b>0.7943</b>	0.8450	0.8246	0.8567	<b>0.8849</b>	<b>0.8236</b>	0.8567	<b>0.8849</b>
	Working time	<b>0.8233</b>	0.8409	<b>0.8006</b>	0.8463	<b>0.8226</b>	0.8566	<b>0.8842</b>	<b>0.8218</b>	0.8566	0.8851
	Todos juntos	<b>0.8323</b>	<b>0.8195</b>	<b>0.7635</b>	0.8276	<b>0.8218</b>	0.8576	0.8853	<b>0.8238</b>	0.8567	<b>0.8837</b>
RMSE	Interpersonal	1.1254	1.1430	1.1547	1.1433	<b>1.1223</b>	<b>1.1620</b>	1.1591	<b>1.1220</b>	<b>1.1620</b>	<b>1.1589</b>
	Emotions	1.1253	1.1417	1.1535	1.1440	<b>1.1228</b>	<b>1.1621</b>	<b>1.1588</b>	<b>1.1217</b>	<b>1.1620</b>	<b>1.1588</b>
	Academic	1.1243	1.1410	1.1534	1.1418	<b>1.1215</b>	1.1623	1.1590	<b>1.1226</b>	1.1623	1.1590
	Parts of day	1.1251	1.1418	1.1530	1.1413	1.1268	1.1624	1.1597	1.1256	1.1624	<b>1.1587</b>
	Working time	1.1280	1.1409	1.1528	1.1437	<b>1.1235</b>	1.1623	<b>1.1589</b>	<b>1.1227</b>	1.1623	<b>1.1589</b>
	Todos juntos	1.1437	1.1387	1.2215	1.1460	<b>1.1230</b>	<b>1.1620</b>	<b>1.1578</b>	<b>1.1233</b>	<b>1.1622</b>	<b>1.1549</b>

**Tabla 5.33** Resultados en la predicción de rating sobre películas empleando métodos conscientes del contexto.

Como era de esperar, no todos los métodos de recomendación contextual resultan más efectivos que los métodos tradicionales, no obstante, los resultados, aunque discretos, son prometedores para estos datos anotados sobre el dominio de películas. De ellos podemos extraer las siguientes conclusiones:

- Los métodos de factorización de matrices CAMF\_C y especialmente CAMF\_CU producen menor error medio absoluto (MAE) que el mejor de los métodos tradicionales (BiasedMF). Cabe destacar, que para el conjunto de datos que tiene todos los contextos juntos, los métodos de factorización menos efectivos como son CAMF\_CI y CAMF\_CUCI ofrecen, de manera significativa, menor MAE en comparación con los conjuntos de datos que tenían en cuenta los contextos por separado.
- Desde el punto de vista de error cuadrático medio (RMSE), los métodos de factorización de matrices (CAMF) no ofrecen buenos resultados, ya que todos ellos cometen más error que el mejor de los métodos base (ItemAverage) e incluso que mayor error que el método de factorización de matrices tradicional.
- Para este conjunto de datos, aplicar las técnicas de *Splitting* (tanto *Item Splitting* como *UI Splitting*) ofrecen buenos resultados. Se puede observar como

si se aplican sobre el mejor de los métodos base (BiasedMF), en la mayoría de los conjuntos de datos los resultados producen menor error tanto MAE como RMSE que empleando dicho método tradicional por sí solo. Revisando también los otros métodos tradicionales (ItemKNN y UserKNN) se puede observar una mejoría para algunos conjuntos de datos frente a los métodos tradicionales, e incluso un efecto similar en otros conjuntos. Esto último es importante, ya que es posible que variando los parámetros a la hora de hacer el *splitting* (e.g., empleando otro criterio de impureza) se obtuvieran mejores resultados que los métodos tradicionales.

Con todo esto, aunque la mejora obtenida en algunos casos sea discreta, se puede decir que, para los datos anotados de películas, aplicar algoritmos de recomendación conscientes del contexto resulta más beneficioso que aplicar métodos de recomendación que no tienen en cuenta en el contexto, por tanto, esto implica que el contexto anotado de manera automática representa una mejora frente al conjunto de datos original.

### 5.3.2 Recomendación sobre música

A continuación, se muestra los resultados obtenidos al aplicar los métodos de referencia o *baseline* sobre los conjuntos de datos anotados de música. De la misma forma que antes, al no tener en cuenta los valores contextuales, estos resultados son los mismos aplicados sobre cualquiera de los 6 conjuntos de datos. Se han resaltado los valores de menor MAE y RMSE para poder compararlos después con los métodos que si tienen en cuenta la información contextual.

<i>Baseline</i>	GlobalAverage	UserAverage	ItemAverage	UserKNN	ItemKNN	BiasedMF
<i>MAE</i>	0.7957	0.7751	0.7343	0.7939	0.7759	<b>0.7226</b>
<i>RMSE</i>	1.0511	1.0828	1.0338	1.0526	1.0556	<b>1.0179</b>

**Tabla 5.34** Resultados en la predicción de rating sobre música empleando métodos *baseline*.

Para este conjunto de datos, los menores resultados en cuanto a menor MAE y menor RMSE se consiguen ambos con el método de factorización de matrices (BiasedMF).

En la siguiente tabla, se muestran los resultados tras evaluar cada uno de los métodos conscientes del contexto sobre cada uno de los 6 conjuntos de datos generados con los datos anotados de música, utilizando los distintos valores contextuales indicados al inicio de la sección, y la unión de todos ellos.

De la misma manera que antes, se han resaltado aquellos resultados con los cuales podemos asegurar que la información contextual añadida en la recomendación resulta beneficiosa. Esto se puede obtener de dos maneras, la **primera** de ellas es obtener menor MAE o RMSE que los mejores casos de los valores de referencia o *baseline*, y la **segunda** de ellas es que los métodos que emplean *Item Splitting* o *UI Splitting*, den mejores resultados que sus métodos de referencia equivalentes, ya que esto también

indica que la información contextual ha sido aprovechada en beneficio de la recomendación, en este aspecto también se han **resaltado** aquellos resultados que tras aplicar el *splitting* producen los mismos resultados que si no se aplicase la separación.

		CAMF C	CAMF CI	CAMF CU	CAMF CUCI	IS + BiasedMF	IS + ItemKNN	IS + UserKNN	UIS + BiasedMF	UIS + ItemKNN	UIS + UserKNN
MAE	Interpersonal	<b>0.7223</b>	<b>0.7217</b>	<b>0.6866</b>	0.7463	<b>0.7213</b>	0.7759	<b>0.7938</b>	<b>0.7217</b>	0.7759	0.7939
	Emotions	0.7248	0.7235	<b>0.6866</b>	0.7446	<b>0.7215</b>	0.7759	0.7939	<b>0.7215</b>	0.7760	0.7939
	Academic	0.7273	0.7282	<b>0.6868</b>	0.7438	<b>0.7213</b>	0.7759	0.7939	<b>0.7221</b>	0.7759	0.7939
	Parts of day	0.7249	0.7273	<b>0.6855</b>	0.7436	<b>0.7220</b>	0.7761	0.7939	<b>0.7225</b>	0.7761	0.7939
	Working time	0.7275	0.7236	<b>0.6906</b>	0.7405	<b>0.7222</b>	0.7760	0.7939	<b>0.7215</b>	0.7760	0.7939
	Todos juntos	0.7531	<b>0.6969</b>	<b>0.6714</b>	0.7310	<b>0.7216</b>	0.7769	<b>0.7938</b>	<b>0.7223</b>	0.7772	0.7939
RMSE	Interpersonal	1.0191	1.0315	1.0437	1.0312	<b>1.0173</b>	1.0556	<b>1.0524</b>	<b>1.0169</b>	<b>1.0555</b>	1.0526
	Emotions	1.0193	1.0333	1.0431	1.0310	<b>1.0166</b>	1.0556	1.0526	<b>1.0175</b>	1.0556	<b>1.0525</b>
	Academic	1.0220	1.0316	1.0439	1.0292	<b>1.0169</b>	1.0556	1.0526	<b>1.0177</b>	1.0556	1.0526
	Parts of day	1.0194	1.0310	1.0439	1.0292	<b>1.0172</b>	1.0558	1.0527	<b>1.0176</b>	1.0558	1.0527
	Working time	1.0216	1.0310	1.0423	1.0273	1.0182	1.0557	1.0527	1.0180	1.0557	1.0527
	Todos juntos	1.0301	1.0276	1.1111	1.0448	<b>1.0148</b>	1.0561	<b>1.0524</b>	<b>1.0176</b>	1.0563	<b>1.0525</b>

**Tabla 5.35** Resultados en la predicción de rating sobre música empleando métodos conscientes del contexto.

Para los datos anotados de música, los resultados quizá no sean tan visibles comparados con el conjunto de datos de películas explicado anteriormente, no obstante podemos extraer las siguientes conclusiones:

- Aplicar las técnicas de *Splitting* puede resultar beneficioso, por un lado, los métodos basados en vecinos más próximos (ItemKNN y UserKNN) prácticamente siguen manteniendo el error cometido frente a no aplicar estas técnicas de *splitting*, por lo que es posible que, variando los parámetros, como por ejemplo el criterio de impureza empleado para la separación, se obtengan mejores resultados que si no realizásemos la división. Por otro lado, aplicadas sobre el método de factorización de matrices tradicional (BiasedMF), el cual era el mejor método de referencia, los resultados son mejores frente a no tener en cuenta el contexto.
- Solo el método de factorización de matrices que emplea la información contextual desde el punto de vista de los usuarios (CAMF\_CU) ofrece un menor error medio absoluto (MAE) que el mejor de los métodos de referencia. Mención especial para el conjunto de datos *Interpersonal*, el cual genera un menor error medio empleando también otros métodos como CAMF\_C o CAMF\_CI.

- Al igual que pasaba con los datos de películas, los métodos de factorización de matrices cometen un mayor error cuadrático medio (RMSE) en todas sus variantes y para todos los conjuntos de datos.

Estos resultados son algo más discretos que para el conjunto de datos del dominio de películas, no obstante, los resultados son prometedores ya que con pequeñas variaciones de en los parámetros de los métodos contextualizados (e.g., técnicas de *splitting*) seguramente obtuviésemos unos resultados mejores que los actuales, por tanto, en este caso también podemos decir que resulta beneficioso aplicar métodos de recomendación conscientes del contexto frente a métodos que no lo tienen en cuenta, por lo que, al igual que pasaba antes, podemos decir que los datos contextuales anotados permiten obtener esto mejores resultados frente a emplear los datos originales.

### 5.3.3 Recomendación sobre libros

En la siguiente tabla se muestran los resultados de los métodos de referencia o *baseline* aplicados sobre el conjunto de datos anotados de libros. De igual manera que antes, se han resaltado los valores de menor MAE y RMSE para poder comparar después con los métodos que si tienen en cuenta la información contextual asociada a los datos.

<i>Baseline</i>	GlobalAverage	UserAverage	ItemAverage	UserKNN	ItemKNN	BiasedMF
<i>MAE</i>	0.8891	0.8774	<b>0.8031</b>	0.8889	0.8751	0.8041
<i>RMSE</i>	1.1269	1.146	<b>1.0636</b>	1.1283	1.1339	1.0750

**Tabla 5.36** Resultados en la predicción de rating sobre libros empleando métodos *baseline*.

Para este conjunto de datos, los menores resultados en cuanto a menor MAE y menor RMSE se consiguen ambos utilizando la media de los ítems directamente (ItemAverage).

A continuación se muestran los resultados tras aplicar los métodos conscientes del contexto sobre los 6 conjuntos de datos asociados a este dominio que tienen en cuenta los distintos valores contextuales.

Se han resaltado aquellos resultados que permiten asegurar que la información contextual añadida mejora los resultados obtenidos en la recomendación. Esto se puede obtener de dos maneras, la **primera** de ellas es obtener menor MAE o RMSE que los mejores casos de los valores de referencia o *baseline*, y la **segunda** de ellas es que los métodos que emplean *Item Splitting* o *UI Splitting*, den mejores resultados que sus métodos de referencia equivalentes, ya que esto también indica que la información contextual ha sido aprovechada en beneficio de la recomendación, en este aspecto también se han **resaltado** aquellos resultados que tras aplicar el *splitting* producen los mismos resultados que si no se aplicase la separación.

		CAMF C	CAMF CI	CAMF CU	CAMF CUCI	IS + BiasedMF	IS + ItemKNN	IS + UserKNN	UIS + BiasedMF	UIS + ItemKNN	UIS + UserKNN
MAE	Interpersonal	0.8057	0.8318	<b>0.7846</b>	0.8367	<b>0.8030</b>	0.8753	<b>0.8888</b>	<b>0.8024</b>	0.8753	<b>0.8888</b>
	Emotions	0.8088	0.8285	<b>0.7907</b>	0.8309	<b>0.8034</b>	0.8751	<b>0.8888</b>	0.8044	0.8751	<b>0.8888</b>
	Academic	0.8073	0.8303	<b>0.7887</b>	0.8314	<b>0.8038</b>	0.8751	0.8889	0.8045	0.8751	0.8889
	Parts of day	0.8092	0.8307	<b>0.7910</b>	0.8318	0.8041	0.8752	0.8889	0.8048	0.8752	0.8889
	Working time	0.8091	0.8312	<b>0.7906</b>	0.8319	<b>0.8038</b>	0.8752	0.8889	0.8044	0.8752	0.8889
	Todos juntos	0.8056	0.8149	<b>0.7773</b>	0.8052	<b>0.8036</b>	0.8753	0.8889	<b>0.8034</b>	0.8752	0.8889
RMSE	Interpersonal	1.0766	1.0968	1.1069	1.0961	<b>1.0730</b>	<b>1.1338</b>	<b>1.1282</b>	<b>1.0734</b>	<b>1.1337</b>	<b>1.1282</b>
	Emotions	1.0775	1.0962	1.1083	1.0948	<b>1.0729</b>	<b>1.1338</b>	1.1283	<b>1.0747</b>	<b>1.1338</b>	1.1283
	Academic	1.0786	1.0955	1.1067	1.0951	<b>1.0735</b>	<b>1.1338</b>	1.1283	1.0752	<b>1.1338</b>	1.1283
	Parts of day	1.0779	1.0962	1.1078	1.0935	1.0763	1.1340	1.1284	1.0765	1.1340	1.1284
	Working time	1.0774	1.0952	1.1091	1.0946	1.0755	1.1340	1.1283	1.0771	1.1340	1.1283
	Todos juntos	1.0828	1.0949	1.1829	1.0885	<b>1.0743</b>	<b>1.1337</b>	1.1284	1.0752	<b>1.1336</b>	1.1284

**Tabla 5.37** Resultados en la predicción de rating sobre libros empleando métodos conscientes del contexto.

Para los datos anotados sobre las revisiones textuales de libros, los resultados son ligeramente menos prometedores que para los dominios anteriores; aun así, se pueden observar puntos positivos al utilizar las anotaciones contextuales frente a no emplearlas.

- Los métodos basados en factorización de matrices no obtienen mejores resultados que el mejor de los métodos de referencia. Únicamente CAMF\_CU mejora los resultados en cuando a obtener un menor error medio absoluto (MAE) que el valor de referencia. De la misma forma que antes, todos ellos producen mayor error cuadrático medio (RMSE) que el valor base.
- Los métodos que realizar *Splitting*, si bien no obtienen mejores resultados que el método de referencia, si mejoran (o se mantienen igual) los métodos tradicionales que no tenían en cuenta la información contextual, por lo que es posible que otras configuraciones de estos métodos (e.g., otro criterio de impureza) pueda ofrecer todavía menor error que los mostrados anteriormente.

En este caso, los datos anotados de contexto ofrecen unos resultados menos visibles frente a los dominios de datos analizados anteriormente (películas y música), de todas formas, se puede observar como las técnicas de *splitting* mejoran los métodos tradicionales que no tenían en cuenta el contexto, por lo que en este caso, aunque resulten menos visibles que antes, los datos anotados de contexto de manera automática también ofrecen una mejora frente a los datos tradicionales.



## Capítulo 6

# Conclusiones y trabajo futuro

En este capítulo se presentan las conclusiones obtenidas tras la realización de este trabajo, junto con distintas líneas de trabajo futuro por las que podría llevarse a cabo una ampliación de lo anteriormente expuesto.

### 6.1 Conclusiones

Tras realizar este trabajo, se han alcanzado una serie de conclusiones que se exponen a continuación:

- Durante el proceso de evaluación manual de las anotaciones a través de la herramienta de validación se ha observado como en la mayor parte de las reviews textuales los usuarios están describiendo el contenido del ítem en lugar de expresar el contexto que rodea el uso del mismo, por lo que ni siquiera realizando una anotación manual se pueden alcanzar grandes cotas de contextos anotados. También se ha visto cómo en las anotaciones revisadas de los dominios de películas, música y libros, la dimensión contextual más presente, desde el punto de vista de anotaciones contextuales correctas, es la dimensión social del usuario (*Social\_context*), el cual principalmente está formado por la compañía social (e.g., con quien vieron una determinada película, o a quien iba destinado cierto libro) y por las emociones (e.g., que les produjo escuchar cierto disco o que sintieron viendo determinada película); junto con la dimensión del tiempo (*Time\_context*) que hace referencia al momento temporal en el que el usuario ha disfrutado del ítem (e.g., cuando vieron cierta película o leyeron cierto libro). No obstante, todo esto está estrechamente ligado con el dominio de

los datos analizadas, ya que es posible que en otros dominios (e.g., críticas de restaurantes u hoteles), los usuarios sean más dados a incluir valores contextuales en sus reviews.

- En el análisis realizado se ha podido comprobar que prácticamente un tercio de las anotaciones contextuales realizadas en los distintos dataset se llevan a cabo a través de sinónimos de la propia categoría, o de sinónimos de instancia, y más tarde en la evaluación manual se puede comprobar que este tipo de anotaciones empleando sinónimos no son efectivas, en primer lugar (Tabla 5.22), se puede observar como las anotaciones realizadas con sinónimos de instancia tienen cotas de acierto extremadamente bajas en comparación con el resto de tipo de anotaciones, y posteriormente (Tabla 5.24) se puede ver como las anotaciones de contexto correctas realizadas por sinónimos producen resultados pobres en comparación con las anotaciones realizadas por instancias. No obstante, tras la validación manual, se ha observado como algunas instancias están produciendo anotaciones que directamente son incorrectas, por lo que sería conveniente poder realizar algún tipo de filtrado sobre las instancias existentes en las distintas categorías de la taxonomía, para no emplear aquellas instancias que producen anotaciones incorrectas.
- Como se ha podido comprobar en la Sección 5.3, los resultados de los métodos de recomendación conscientes del contexto suelen mejorar los resultados de los métodos de referencia, ya sea de forma directa generando menor error que el mejor de los métodos *baseline* o si hablamos de *Item Splitting* o *UI Splitting*, ofreciendo mejores resultados en comparación con el método de referencia antes de aplicar la técnica de *splitting*. Por esto, queda constatado que cualquier información contextual es beneficiosa a la hora de realizar la recomendación.
- Las evaluaciones se han centrado en términos de predicción de rating (MAE y RMSE) debido a que dada la distribución de las reviews empleada para realizar este trabajo (100k datos de cada dominio), son pocos los usuarios que tienen varios ratings, por lo que para llevar a cabo una evaluación desde el punto de vista de predicción de ranking, sería necesario realizar el proceso con un número mayor de datos, de forma que haya una mayor cantidad de usuarios que tengan varios ratings para poder establecer así cuál sería su ranking ideal de ítems.



## 6.2 Trabajo futuro

Durante la realización de esta investigación se han identificado una serie de líneas de trabajo futuro que podrían llevarse a cabo con el fin de realizar una ampliación. A continuación se exponen dichas ideas:

- Como una gran parte de las reviews describen el contenido del ítem en lugar del contexto, se propone emplear otros dominios dentro del dataset de Amazon, u otro conjunto de datos, como puede ser el dataset de Yelp<sup>16</sup>, que contiene reviews de distintos tipos de negocios como por ejemplo restaurantes, donde los usuarios son más propensos a expresar el contexto (e.g., con quien fuero al restaurante, en qué momento acudieron, etc.) con el fin de realizar un mayor número de anotaciones contextuales correctas.
- Como se ha visto, las anotaciones realizadas por sinónimos a nivel general son incorrectas en la mayoría de los casos, por lo que se propone realizar el anotado contextual sin tener en cuenta los sinónimos de la categoría contextual, los sinónimos de las instancias, ni los sinónimos de la redirección.
- Con el fin de poder llevar a cabo una revisión manual más exhaustiva, se propone enfocar esta evaluación de anotaciones como un proyecto *crowdsourcing* de forma que un mayor número de usuarios puedan formar parte del proceso de evaluación de las anotaciones con el fin de aumentar la cobertura de la misma.
- Respecto a los algoritmos de recomendación, se hace necesario realizar un estudio más profundo, como por ejemplo teniendo en cuenta los siguientes puntos:
  - Se propone experimentar empleando el método DSPF (*Distributional-Semantics Pre-Filtering*) desarrollado por Codina et al. (2016), ya que, en vista de su estudio este método ofrece mejores resultados (MAE y nDCG) que otros algoritmos considerados como el estado del arte, como son CAMF (*Context-Aware Matrix Factorization*) y *UI Splitting*.
  - A la vista de los resultados al emplear las técnicas *Item Splitting* y *UI Splitting*, donde en gran parte de los casos mejoran los resultados de los métodos de referencia (o los igualan) se propone emplear otros criterios de impureza como pueden ser el criterio de divergencia de la información (*Information Gain*) o el criterio *Chi Squared* descrito ambos por Baltraunas et al. (2014).

---

<sup>16</sup> Dataset de Yelp, reviews de distintos negocios, [https://www.yelp.com/dataset\\_challenge](https://www.yelp.com/dataset_challenge)

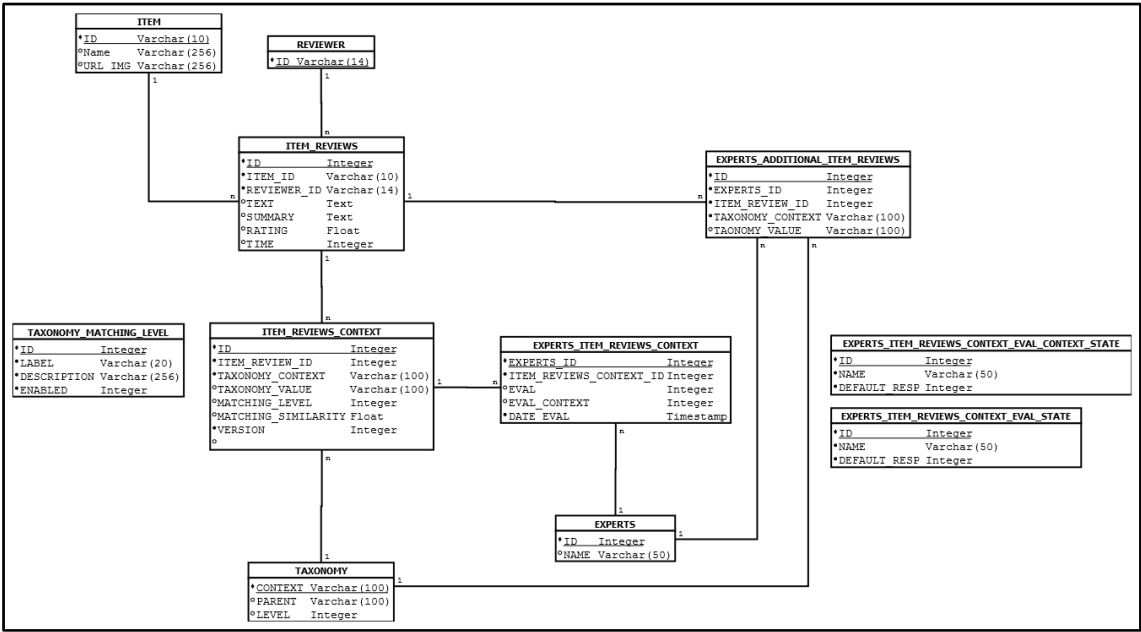
- En lugar de aplicar validación cruzada, generar un conjunto de test fijo con el que probar los métodos de recomendación. Para ello se propone seleccionar aquellos usuarios que tengan como mínimo un cierto número de ratings y aquellos ítems con cierto número de reviews mínimo, y de ese conjunto seleccionar al azar un rating por cada usuario para emplearlo en el conjunto de test y dejar el resto para el conjunto de entrenamiento. De esta forma se evita el problema de *Cold Start* o arranque en frío causado por la falta de información de los usuarios.
- Realizar una evaluación desde el punto de vista de predicción de ranking en lugar de evaluar la predicción de rating. Para ello es necesario emplear un mayor número de datos para así tener un mayor número de usuarios con múltiples ratings con los que poder establecer cuál sería el ranking ideal de ítems desde su punto de vista.

# Anexo A

## Base de datos de la aplicación de evaluación de anotaciones

A continuación se detalla el diseño de base de datos utilizado para la aplicación de evaluación de anotaciones contextuales. Este diseño permite cargar las reviews con anotaciones contextuales (junto con algunos metadatos de los ítems), para que sean revisadas por parte de los usuarios (denominados *expertos*); y a su vez permite que los usuarios puedan incluir anotaciones contextuales de manera manual que no hayan sido anotadas de forma automática.

En la siguiente figura se muestra el diagrama Entidad-Relación de las distintas tablas existentes:



**Figura A.1** Diagrama Entidad-Relación de la base de datos para la aplicación de evaluación de anotaciones contextuales

Como se puede observar, el diseño cuenta con un total de 11 tablas, cuya estructura se detalla a continuación:

**Tabla: EXPERTS**

Esta tabla incluye la información de los usuarios o expertos que se encargan de evaluar las anotaciones contextuales.

Columna	Tipo	Nulo	Predeterminado	Comentario
<u>ID</u>	<i>Integer</i>	No		ID del experto que revisa las anotaciones
NAME	<i>Varchar(50)</i>	Si	<i>Null</i>	Nombre del experto que revisa las anotaciones

**Tabla A.1** Campos de la tabla EXPERTS

**Tabla: EXPERTS\_ADDITIONAL\_ITEM\_REVIEWS**

Esta tabla sirve para almacenar las anotaciones manuales encontradas por parte de los usuarios dentro del texto de la review, indicando la palabra dentro de la review y el valor contextual de la taxonomía la que pertenece.

Columna	Tipo	Nulo	Predeterminado	Comentario
<u>ID</u>	<i>Integer</i>	No		ID autogenerado del contexto adicional incluido por el experto.
EXPERTS_ID	<i>Integer</i>	No		ID del experto que realizó la anotación manual.
ITEM_REVIEW_ID	<i>Integer</i>	No		ID de la review a la que pertenece el contexto anotado manualmente.
TAXONOMY_CONTEXT	<i>Varchar(100)</i>	No		ID del contexto dentro de la taxonomía al que pertenece la anotación.
TAXONOMY_VALUE	<i>Varchar(100)</i>	Si	<i>Null</i>	Palabra identificada por el experto dentro de la review como contexto.

**Tabla A.2** Campos de la tabla EXPERTS\_ADDITIONAL\_ITEM\_REVIEWS

**Tabla: EXPERTS\_ITEM\_REVIEWS\_CONTEXT**

Esta tabla sirve para almacenar las distintas evaluaciones realizadas por los usuarios de las anotaciones contextuales, almacenando por un lado si el contexto identificado es en verdad contexto, y por otro lado si el *matching* realizado es correcto.

Columna	Tipo	Nulo	Predeterminado	Comentario
<u>EXPERTS_ID</u>	Integer	No		ID del experto que realizó la evaluación
<u>ITEM_REVIEWS_CONTEXT_ID</u>	Integer	No		ID de la anotación contextual evaluada.
EVAL	Integer	Si	Null	Evaluación de la anotación.
EVAL_CONTEXT	Integer	Si	Null	Evaluación del contexto anotado.
DATE_EVAL	Timestamp	No	Current_timestamp	Fecha en la que el experto realizó la evaluación.

**Tabla A.3** Campos de la tabla EXPERTS\_ITEM\_REVIEWS\_CONTEXT**Tabla: EXPERTS\_ITEM\_REVIEWS\_CONTEXT\_EVAL\_CONTEXT\_STATE**

Esta tabla representa el catálogo de estados para validar únicamente la parte del contexto anotado, es decir, si el contexto anotado es o no contexto, si es contexto, pero se refiere a la descripción del ítem en vez de al usuario, etc. También se puede determinar cuál es el estado por defecto empleado para la anotación, dicho estado se mostrará el primero en el desplegable para validar dentro de la aplicación.

Columna	Tipo	Nulo	Predeterminado	Comentario
<u>ID</u>	Integer	No		ID del estado para evaluar el contexto anotado
NAME	Varchar(50)	No		Descripción del estado para evaluar el contexto
DEFAULT_RESP	Integer	No	0	1 si es el estado por defecto, 0 si no.

**Tabla A.4** Campos de la tabla EXPERTS\_ITEM\_REVIEWS\_CONTEXT\_EVAL\_CONTEXT\_STATE

**Tabla: EXPERTS\_ITEM\_REVIEWS\_CONTEXT\_EVAL\_STATE**

Esta tabla representa el catálogo de estados para validar la anotación en sí misma, es decir, para determinar si es correcta, si es incorrecta, si es correcta, pero hubiese sido mejor un nivel inferior/superior de la taxonomía, etc.

Columna	Tipo	Nulo	Predeterminado	Comentario
<i>ID</i>	<i>Integer</i>	No		ID del estado para evaluar la anotación
<i>NAME</i>	<i>Varchar(50)</i>	No		Descripción del estado para evaluar la anotación.
<i>DEFAULT_RESP</i>	<i>Integer</i>	No	<i>0</i>	1 si es el estado por defecto, 0 si no.

**Tabla A.5** Campos de la tabla EXPERTS\_ITEM\_REVIEWS\_CONTEXT\_EVAL\_STATE**Tabla: ITEM**

Esta tabla contiene los metadatos de los ítems a los que pertenecen las reviews.

Columna	Tipo	Nulo	Predeterminado	Comentario
<i>ID</i>	<i>Varchar(10)</i>	No		ID del ítem
<i>NAME</i>	<i>Varchar(256)</i>	Si	<i>Null</i>	Nombre del ítem
<i>URL_IMG</i>	<i>Varchar(256)</i>	Si	<i>Null</i>	URL con la imagen del ítem

**Tabla A.6** Campos de la tabla ITEM**Tabla: ITEM\_REVIEWS**

Esta tabla contiene las reviews textuales realizadas por parte de los usuarios que han consumido los ítems, así como el rating asignado por ellos.

Columna	Tipo	Nulo	Predeterminado	Comentario
<i>ID</i>	<i>Integer</i>	No		ID autogenerado para la review del usuario sobre el ítem.
<i>ITEM_ID</i>	<i>Varchar(10)</i>	No		ID del ítem al que hace referencia la review
<i>REVIEWER_ID</i>	<i>Varchar(14)</i>	No		ID del usuario que realizó la review
<i>TEXT</i>	<i>Text</i>	Si	<i>Null</i>	Texto completo de la review introducido por el usuario
<i>SUMMARY</i>	<i>Text</i>	Si	<i>Null</i>	Resumen para la review introducido por el usuario
<i>RATING</i>	<i>Float</i>	Si	<i>Null</i>	Rating asignado al ítem por el usuario
<i>TIME</i>	<i>Integer</i>	Si	<i>Null</i>	Tiempo en el que se realizó la review

**Tabla A.7** Campos de la tabla ITEM\_REVIEWS

**Tabla: ITEM\_REVIEWS\_CONTEXT**

Esta tabla almacena las distintas anotaciones contextuales realizadas de manera automática sobre las reviews de los usuarios, indicando el nivel de matching empleado, y el contexto identificado dentro de la taxonomía.

Columna	Tipo	Nulo	Predeterminado	Comentario
<u>ID</u>	Integer	No		ID autogenerado para la anotación contextual.
ITEM_REVIEW_ID	Integer	No		ID de la review a la que pertenece la anotación
TAXONOMY_CONTEXT	Varchar(100)	No		ID del contexto dentro de la taxonomía al que pertenece la anotación
TAXONOMY_VALUE	Varchar(100)	Si	Null	Palabra identificada dentro del texto de la review como contexto
MATCHING_LEVEL	Integer	Si	Null	Nivel de matching empleado: si es un concepto, una instancia, un sinónimo, etc.
MATCHING_SIMILARITY	Float	Si	Null	Similitud entre el texto identificado y el contexto de la taxonomía
VERSION	Integer	No	1	Versión de la anotación contextual, por si existiesen distintas versiones de algoritmos de anotado.

**Tabla A.8** Campos de la tabla ITEM\_REVIEWS\_CONTEXT**Tabla: REVIEWER**

Esta tabla sirve para almacenar los metadatos de los usuarios que realizaron la review del ítem. De momento cuenta solo con el ID, pero podría incluirse todo tipo de información del usuario como su nombre, procedencia, gustos, etc.

Columna	Tipo	Nulo	Predeterminado	Comentario
<u>ID</u>	Varchar(14)	No		ID del usuario que realizó la review del ítem

**Tabla A.9** Campos de la tabla REVIEWER

**Tabla: TAXONOMY**

Esta tabla sirve para almacenar la taxonomía de contexto empleada durante el proceso de anotación, y que sirve durante el proceso de validación como apoyo para el usuario tanto en la tarea de validar las anotaciones existentes, como para incluir las anotaciones manuales.

Columna	Tipo	Nulo	Predeterminado	Comentario
<i><u>CONTEXT</u></i>	<i>Varchar(100)</i>	No		Nodo de la taxonomía que representa contexto
<i>PARENT</i>	<i>Varchar(100)</i>	Si		Padre del nodo de contexto, o <i>null</i> si se trata del nodo raíz de la taxonomía
<i>LEVEL</i>	<i>Integer</i>	Si	<i>Null</i>	Nivel donde se encuentra el contexto dentro de la taxonomía siendo 0 el nodo raíz

**Tabla A.10** Campos de la tabla TAXONOMY

**Tabla: TAXONOMY\_MATCHING\_LEVEL**

Esta tabla contiene el catálogo con los distintos niveles de *matching* que se emplean dentro del proceso de anotación contextual, como puede ser una coincidencia exacta entre la palabra y el contexto, una coincidencia con un sinónimo del contexto, una coincidencia con una instancia del contexto extraída de DBpedia, etc.

Columna	Tipo	Nulo	Predeterminado	Comentario
<i><u>ID</u></i>	<i>Integer</i>	No		ID correspondiente al nivel de matching empleado en las anotaciones
<i>LABEL</i>	<i>Varchar(20)</i>	No		Etiqueta corta para identificar el nivel de matching
<i>DESCRIPTION</i>	<i>Varchar(256)</i>	No		Descripción auxiliar completa para el nivel de matching
<i>ENABLED</i>	<i>Integer</i>	No	<i>1</i>	1 si se quiere mostrar dentro de la tabla auxiliar de la taxonomía que se muestra en la aplicación, 0 si no.

**Tabla A.11** Campos de la tabla TAXONOMY\_MATCHING\_LEVEL



# Referencias

- Abowd , G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., & Steggles, P. (1999). Towards a better understanding of context and context-awareness. In *Proceedings of the International Symposium on Handheld and Ubiquitous Computing*, pp. 304-307.
- Aciar, S. (2010, September). Mining context information from consumers reviews. In *Proceedings of the 2nd Workshop on Context-Aware Recommender System*.
- Aciar, S., Zhang, D., Simoff, S., & Debenham, J. (2007). Informed recommender: Basing recommendations on consumer product reviews. *IEEE Intelligent Systems* 22(3), 39-47.
- Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6), 734-749.
- Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In *Recommender Systems Handbook*, pp. 217-253. Springer.
- Adomavicius, G., Mobasher, B., Ricci, F., Tuzhilin, A. (2011) Context-aware recommender systems. *AI Magazine* 32(3), pp. 67-80.
- Adomavicius, G., Sankaranarayanan, R., Sen, S., & Tuzhilin, A. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems* 23(1), 103-145.
- Baeza-Yates, R., & Ribeiro-Neto, B. (1999). *Modern information retrieval*. ACM Press Books.
- Baldauf, M., Dustdar, S., & Rosenberg, F. (2007). A survey on context-aware systems. *International Journal of Ad Hoc and Ubiquitous Computing* 2(4), 263-277.
- Baltrunas, L., & Ricci, F. (2009). Context-based splitting of item ratings in collaborative filtering. In *Proceedings of the 3rd ACM Conference on Recommender Systems*, pp. 245-248.
- Baltrunas, L., & Ricci, F. (2014). Experimental evaluation of context-dependent collaborative filtering using item splitting. *User Modeling and User-Adapted Interaction* 24(1-2), 7-34.
- Baltrunas, L., Ludwig, B., & Ricci, F. (2011). Matrix factorization techniques for context aware recommendation. In *Proceedings of the 5th ACM Conference on Recommender Systems*, pp. 301-304.

- Bauman, K., & Tuzhilin, A. (2014). Discovering contextual information from user reviews for recommendation purposes. In *Proceedings of the 1st Workshop on New Trends in Content-based Recommender Systems*, pp. 2-9.
- Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., & Riboni, D. (2010). A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing* 6(2), 161-180.
- Burke, R.D. (2002) Hybrid recommender systems: survey and experiments. *User Modeling and User-Adapted Interaction* 12(4), 331-370.
- Castells, P. (2003). La web semántica. In *Sistemas Interactivos y Colaborativos en la Web*, pp. 195-212.
- Chen, G., & Kotz, D. (2000). A survey of context-aware mobile computing research. *Technical Report TR2000-381, Department of Computer Science, Dartmouth College*.
- Codina, V., Ricci, F., & Ceccaroni, L. (2016). Distributional semantic pre-filtering in context-aware recommender systems. *User Modeling and User-Adapted Interaction* 26(1), 1-32.
- Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3), 171-176.
- Dey, A.K. (2001) Understanding and using context. *Personal and Ubiquitous Computing* 5(1), 4-7.
- Ejigu, D., Scuturici, M., & Brunie, L. (2007). An ontology-based approach to context modeling and reasoning in pervasive computing. In *Proceedings of the 5th Annual IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 14-19.
- Gu, T., Pung, H. K., & Zhang, D. Q. (2005). A service-oriented middleware for building context-aware services. *Journal of Network and Computer Applications* 28(1), 1-18.
- Hariri, N., Zheng, Y., Mobasher, B., & Burke, R. (2011). Context-aware recommendation based on review mining. In *Proceedings of the 9th Workshop on Intelligent Techniques for Web Personalization & Recommender Systems*.
- Hong, J. Y., Suh, E. H., & Kim, S. J. (2009). Context-aware systems: a literature review and classification. *Expert Systems with Applications* 36(4), 8509-8522.
- Hoseini-Tabatabaei, S. A., Gluhak, A., & Tafazolli, R. (2013). A survey on smartphone-based systems for opportunistic user context recognition. *ACM Computing Surveys* 45(3), art. 27

- Karatzoglou, A., Amatriain, X., Baltrunas, L., & Oliver, N. (2010). Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the 4th ACM conference on Recommender systems*, pp. 79-86.
- Kim, J. D., Son, J., & Baik, D. K. (2012). CA5W1H Onto: ontological context-aware model based on 5W1H. *International Journal of Distributed Sensor Networks*, vol. 2012.
- Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8).
- Krummenacher, R., & Strang, T. (2007). Ontology-based context modeling. In *Proceedings of the 3rd Workshop on Context-Aware Proactive Systems*.
- Lahlou, F. Z., Benbrahim, H., Mountassir, A., & Kassou, I. (2013). Inferring context from users' reviews for context aware recommendation. In *Research and Development in Intelligent Systems XXX*, pp. 227-239.
- Li, Y., Nie, J., Zhang, Y., Wang, B., Yan, B., & Weng, F. (2010). Contextual recommendation based on text mining. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 692-700.
- Linden G., Smith B., York J. (2003). Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing* 7(1), 76-80.
- Levenshtein, V. I. (1966, February). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet Physics Doklady* (Vol. 10, No. 8, pp. 707-710).
- Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 55-60.
- McAuley, J., & Leskovec, J. (2013, October). Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems* (pp. 165-172). ACM.
- Miller, G.A. (1995). WordNet: a lexical database for English. *Communications of the ACM* 38(11), 39-41.
- Palmisano, C., Tuzhilin, A., & Gorgoglione, M. (2008). Using context to improve predictive modeling of customers in personalization applications. *IEEE Transactions on Knowledge and Data Engineering* 20(11), 1535-1549.
- Pazzani, M., & Billsus, D. (1997). Learning and revising user profiles: The identification of interesting web sites. *Machine learning* 27(3), 313-331.

- Preuveneers, D., Van den Bergh, J., Wagelaar, D., Georges, A., Rigole, P., Clerckx, T., ... & De Bosschere, K. (2004). Towards an extensible context ontology for ambient intelligence. In *Proceedings of the 2nd European Symposium on Ambient Intelligence*, pp. 148-159.
- Quinlan, J. R. (2014). *C4. 5: programs for machine learning*. Elsevier.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, pp. 175-186.
- Salton, G. (1989). *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley.
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International on World Wide Web*, pp. 285-295.
- Shani, G., & Gunawardana, A. (2011). Evaluating recommendation systems. In *Recommender Systems Handbook*, pp. 257-297. Springer.
- Strang, T., & Linnhoff-Popien, C. (2004). A Context Modeling Survey. In *Proceedings of the 2004 Workshop on Advanced Context Modelling, Reasoning and Management*.
- Wang, X. H., Zhang, D. Q., Gu, T., & Pung, H. K. (2004). Ontology based context modeling and reasoning using OWL. In *Proceedings of the 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp. 18-22.
- Zheng, Y., Burke, R., & Mobasher, B. (2014). Splitting approaches for context-aware recommendation: An empirical study. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pp. 274-279.
- Zheng, Y., Mobasher, B., & Burke, R. (2015, November). Carskit: A java-based context-aware recommendation engine. In *Data Mining Workshop (ICDMW), 2015 IEEE International Conference on* (pp. 1668-1671). IEEE.